

### บทที่ 3

## การละเมิดลิขสิทธิ์ในโปรแกรมคอมพิวเตอร์โดยการดัดแปลง

ในบทนี้จะกล่าวถึงการละเมิดลิขสิทธิ์ในโปรแกรมคอมพิวเตอร์โดยการดัดแปลง ซึ่งมีความสลับซับซ้อนและยากในการพิจารณามากกว่าการละเมิดลิขสิทธิ์ในโปรแกรมคอมพิวเตอร์ โดยการทำซ้ำ เพราะการทำซ้ำนั้นเป็นสิ่งที่สามารถเห็นได้อย่างชัดเจนว่าได้มีการคัดลอกตัวอักษรที่มีลักษณะเป็นการละเมิดทางตัวอักษร (Literal Infringement) ไม่ว่าจะเป็นการทำซ้ำในโปรแกรมต้นฉบับ (source code) หรือโปรแกรมภาษาเครื่อง(object code)จะเป็นสิ่งที่สามารถพิสูจน์ถึงการละเมิดลิขสิทธิ์ได้ไม่ยาก ซึ่งต่างกับการพิสูจน์ถึงการละเมิดลิขสิทธิ์ในโปรแกรมคอมพิวเตอร์โดยการดัดแปลงเพราะมีลักษณะที่ไม่มีความเหมือนกันในตัวอักษร(Nonliteral) ซึ่งเป็นการลอกเลียนส่วนประกอบอื่นๆ นอกเหนือจากการคัดลอกตัวอักษร เช่น โครงสร้างของโปรแกรม, การจัดลำดับขั้นตอนของโปรแกรมและการจัดวางองค์ประกอบของโปรแกรมคอมพิวเตอร์ และการเหมือนกันในการแสดงออกทางจอภาพ (screen display) ทำให้การพิสูจน์ว่าเป็นการละเมิดลิขสิทธิ์หรือไม่นั้นทำได้ยาก

### 3.1 องค์ประกอบของการดัดแปลงในโปรแกรมคอมพิวเตอร์ตามคำนิยามในมาตรา 4

ในพระราชบัญญัติลิขสิทธิ์ พ.ศ. 2537 ได้ให้คำนิยามของคำว่า “ดัดแปลง” ไว้ในมาตรา 4 ว่า “ดัดแปลง หมายความว่า ทำซ้ำโดยเปลี่ยนรูปแบบ ปรับปรุงแก้ไขเพิ่มเติม หรือจำลองงานต้นฉบับในส่วนอันเป็นสาระสำคัญโดยไม่มีลักษณะเป็นการจัดทำขึ้นใหม่ ทั้งนี้ไม่ว่าทั้งหมด หรือ บางส่วน ....(2) ในส่วนที่เกี่ยวกับโปรแกรมคอมพิวเตอร์ ให้หมายความรวมถึง ทำซ้ำโดยเปลี่ยนรูปแบบ ปรับปรุง แก้ไขเพิ่มเติมโปรแกรมคอมพิวเตอร์ในส่วนอันเป็นสาระสำคัญ โดยไม่มีลักษณะเป็นการจัดทำขึ้นใหม่ ”

ก่อนที่จะกล่าวถึงลักษณะของการดัดแปลงโปรแกรมคอมพิวเตอร์ในลักษณะต่าง ๆ ว่าเป็นการละเมิดลิขสิทธิ์หรือไม่ จำเป็นที่จะต้องทำความเข้าใจถึงองค์ประกอบของการดัดแปลงในโปรแกรมคอมพิวเตอร์ในมาตรา 4 ของพระราชบัญญัติลิขสิทธิ์ พ.ศ. 2537 ซึ่งแบ่งองค์ประกอบได้เป็น 4 ประการคือ

ก. การทำซ้ำโดยเปลี่ยนรูปใหม่หรือปรับปรุง หรือแก้ไขเพิ่มเติมหรือจำลองงานต้นฉบับโปรแกรมคอมพิวเตอร์

ความหมายของการดัดแปลงนั้นหมายถึงการทำซ้ำโดยการเปลี่ยนรูปใหม่ หรือการแก้ไขเพิ่มเติมหรือจำลองงานต้นฉบับโปรแกรมคอมพิวเตอร์ สำหรับการทำซ้ำนั้นหมายถึงการคัดลอกหรือทำสำเนาโปรแกรมคอมพิวเตอร์จากสื่อบันทึกใด ไม่ว่าด้วยวิธีใดๆ ในส่วนอันเป็นสาระสำคัญ โดยไม่มีลักษณะเป็นการจัดทำงานขึ้นมาใหม่ ทั้งนี้ไม่ว่าทั้งหมดหรือบางส่วน ดังนั้นถ้าเป็นการคัดลอกหรือทำสำเนาโปรแกรมคอมพิวเตอร์โดยมีการเปลี่ยนรูปใหม่ หรือการปรับปรุง หรือการแก้ไขเพิ่มเติมโปรแกรมคอมพิวเตอร์ เช่น เดิมมีโปรแกรมหนึ่งเขียนอยู่ในรูปของภาษาคอมพิวเตอร์ภาษาเบสิก ต่อมาได้นำมาดัดแปลงโดยเขียนโปรแกรมโดยใช้ภาษาซี หรือกรณีของโปรแกรมที่ใช้ในการทำบัญชี แล้วต่อมาได้นำโปรแกรมนั้นมาแก้ไขเพิ่มเติมบางส่วนลงไปที่ทำให้การทำงานสะดวกขึ้น กรณีอย่างนี้ถือว่าเป็นการดัดแปลงแล้ว

ข. เป็นการกระทำในส่วนอันเป็นสาระสำคัญของโปรแกรมคอมพิวเตอร์

การดัดแปลงต้องเป็นการกระทำต่อส่วนอันเป็นสาระสำคัญของโปรแกรมคอมพิวเตอร์ ส่วนอันเป็นสาระสำคัญของโปรแกรมนั้นเป็นสิ่งที่ยากในการพิจารณาว่าส่วนใดของโปรแกรมเป็นส่วนอันเป็นสาระสำคัญ ซึ่งในการพิสูจน์อาจจะนำผู้เชี่ยวชาญทางโปรแกรมคอมพิวเตอร์มาพิจารณาว่าส่วนของโปรแกรมที่มีการดัดแปลงเป็นส่วนอันเป็นสาระสำคัญของโปรแกรมหรือไม่ เช่น การดัดแปลงโปรแกรมคอมพิวเตอร์คอมพิวเตอร์จากภาษาโปรแกรมคอมพิวเตอร์หนึ่งไปสู่อีกภาษาหนึ่ง ถือได้ว่าเป็นการดัดแปลงในส่วนอันเป็นสาระสำคัญของโปรแกรมแล้ว

ค. เป็นการกระทำโดยไม่มีลักษณะเป็นการจัดทำงานขึ้นมาใหม่

นอกจากการดัดแปลงต้องเป็นการกระทำต่อส่วนอันเป็นสาระสำคัญของโปรแกรมคอมพิวเตอร์โดยไม่มีลักษณะเป็นการจัดทำงานนั้นขึ้นมาใหม่ เพราะถ้าเป็นการจัดทำงานนั้นขึ้นมาใหม่ผู้ดัดแปลงย่อมมีลิขสิทธิ์ในงานที่ดัดแปลงนั้นและไม่ถือเป็นการกระทำที่ละเมิดลิขสิทธิ์

### ง. การตัดแปลงนั้นได้กระทำไปทั้งหมดหรือบางส่วน

การตัดแปลงไม่จำเป็นต้องตัดแปลงทั้งหมดของโปรแกรมจึงจะถือว่าเป็นการละเมิดลิขสิทธิ์โดยการตัดแปลง แม้จะมีการตัดแปลงเพียงบางส่วนของโปรแกรมแต่การกระทำครบองค์ประกอบที่กฎหมายกำหนดก็ถือว่าเป็นการตัดแปลงตามกฎหมายแล้วเพราะกฎหมายกำหนดว่า “ไม่ว่าทั้งหมดหรือบางส่วน” เช่น อาจตัดแปลงโดยการแปลจากภาษาคอมพิวเตอร์ภาษาหนึ่งเป็นอีกภาษาหนึ่ง หรือ ตัดแปลงในโปรแกรมย่อย(subroutine)บางส่วนของโปรแกรม เป็นต้น แต่ถ้าเป็นการตัดแปลงงานอันมีลิขสิทธิ์โดยได้รับอนุญาตจากเจ้าของลิขสิทธิ์ ผู้ที่ตัดแปลงก็จะมีลิขสิทธิ์ในงานที่ตัดแปลงนั้นด้วยแต่ต้องไม่กระทบกระเทือนสิทธิของเจ้าของลิขสิทธิ์ที่มีอยู่ในงานเดิม ตามที่ได้บัญญัติไว้ในมาตรา 11 ของพระราชบัญญัติลิขสิทธิ์ พ.ศ. 2537

### 3.2 การกระทำที่เป็นการตัดแปลงในโครงสร้าง (structure), การจัดลำดับขั้นตอน (sequence) และการจัดวางองค์ประกอบ (organization) ของโปรแกรมคอมพิวเตอร์

ในหัวข้อการกระทำที่เป็นการละเมิดลิขสิทธิ์ในบทที่ 2 หัวข้อ 2.2.1, 2.2.2, 2.2.3 นั้นเป็นการละเมิดลิขสิทธิ์ในโปรแกรมคอมพิวเตอร์ในลักษณะที่เป็นการทำซ้ำในตัวอักษรไม่ว่าจะเป็นโปรแกรมต้นฉบับ(source code) หรือโปรแกรมภาษาเครื่อง(object code)มีลักษณะไม่แตกต่างไปจากการลอกเลียนทำซ้ำหนังสือ เนื่องจากภาษาที่นักเขียนโปรแกรมใช้ในการเขียนโปรแกรมนั้นมีลักษณะที่คล้ายกับตัวอักษรหนังสือภาษาอังกฤษธรรมดา เพียงแต่ลักษณะของการใช้ภาษา, คำสั่งและเทคนิคจะแตกต่างไปจากการเขียนหนังสือธรรมดา ผู้ที่จะเขียนโปรแกรมคอมพิวเตอร์ได้ต้องเรียนรู้ถึงภาษาที่ใช้ในการเขียนโปรแกรมและเทคนิคการแก้ปัญหาต่าง ๆ การลอกเลียนตัวอักษรและการกระทำที่เป็นการทำซ้ำนั้นไม่มีการซับซ้อนมากนัก การพิจารณาของศาลจึงตัดสินใจได้ง่ายว่าการกระทำใดเป็นการทำซ้ำที่ละเมิดลิขสิทธิ์ในโปรแกรมคอมพิวเตอร์หรือไม่ แม้ว่าการปรับใช้กฎหมายลิขสิทธิ์กับการคุ้มครองโปรแกรมคอมพิวเตอร์ของยากลำบากเมื่อเทียบกับการปรับใช้กฎหมายลิขสิทธิ์กับการคุ้มครองงานวรรณกรรมทั่วไปก็ตาม แต่คำพิพากษาของศาลก็ให้คำตอบที่กระจ่างชัดถึงการคุ้มครองโปรแกรมคอมพิวเตอร์อันสืบเนื่องมาจากการทำซ้ำ

เนื่องจากเทคโนโลยีความรู้ต่าง ๆ เกี่ยวกับโปรแกรมคอมพิวเตอร์ได้เจริญก้าวหน้ามากขึ้น การละเมิดลิขสิทธิ์ในโปรแกรมคอมพิวเตอร์ จึงไม่ได้เป็นเพียงการลอกเลียน ทำซ้ำในลักษณะที่เป็นตัวอักษร ที่ง่ายในการพิสูจน์ถึงการละเมิดลิขสิทธิ์ แต่จะเป็นการกระทำที่เป็นการ

ดัดแปลงโปรแกรมคอมพิวเตอร์ เช่น การลอกเลียนในส่วนของโครงสร้าง, การจัดลำดับขั้นตอน และการจัดวางองค์ประกอบของโปรแกรมหรือที่เรียกว่า structure, sequence และ organization เป็นต้น ซึ่งเป็นการลอกเลียนโดยไม่มีความเหมือนคล้ายกันหรือทำซ้ำในตัวอักษร ดังนั้นในการที่จะพิจารณาว่าเป็นการละเมิดลิขสิทธิ์หรือไม่ ผู้พิพากษาจะต้องเป็นผู้พิจารณาและปรับใช้กฎหมายเพราะเนื่องจากตัวบทกฎหมายไม่ได้บัญญัติถึงสิ่งเหล่านี้ไว้โดยตรง จึงเป็นปัญหาของศาลในการพิจารณาถึงบทบัญญัติของกฎหมายลิขสิทธิ์, ผลประโยชน์ของโจทก์จำเลยและผลประโยชน์ของสังคมส่วนรวม เพราะการขยายขอบเขตของการคุ้มครองที่มากเกินไปนั้นจะทำให้เจ้าของลิขสิทธิ์มีสิทธิแต่เพียงผู้เดียวในงานนั้นโดยที่สังคมส่วนรวมไม่สามารถใช้ประโยชน์จากงานนั้นได้ ในหัวข้อนี้จึงเป็นการพิจารณาว่ากฎหมายลิขสิทธิ์สมควรที่จะขยายการคุ้มครองไปถึงโครงสร้าง, การจัดลำดับขั้นตอนและการจัดวางองค์ประกอบของโปรแกรมหรือไม่

#### ก. กฎหมายของสหรัฐอเมริกา

ประเทศสหรัฐอเมริกานั้นเป็นประเทศที่มีความเจริญก้าวหน้าทางเทคโนโลยีด้านโปรแกรมคอมพิวเตอร์มาก จึงมีการกระทำที่อาจเป็นการละเมิดลิขสิทธิ์เกิดขึ้นในหลายรูปแบบ โดยศาลต้องพิจารณาคัดสินว่าการกระทำใดเป็นการละเมิดลิขสิทธิ์ในโปรแกรมคอมพิวเตอร์หรือไม่ รวมทั้งกรณีของการลอกเลียนในส่วนของโครงสร้าง, การจัดลำดับขั้นตอนและการจัดวางองค์ประกอบของโปรแกรมคอมพิวเตอร์ว่าเป็นการละเมิดลิขสิทธิ์หรือไม่ เพราะการกระทำดังกล่าวไม่มีการคัดลอกหรือทำซ้ำในตัวอักษร (Nonliteral Infringement)

ประเด็นดังกล่าวข้างต้นมีคดีเกิดขึ้นหลายคดีที่ศาลต้องพิจารณา เช่น คดี Synercom Technology Inc. V. University Computing Co.,<sup>1</sup> คดี SAS Institute Inc. V. S&H Computer Systems Inc.,<sup>2</sup> และคดีที่ถือว่าเป็นบรรทัดฐานในการที่ศาลให้การคุ้มครองแก่โครงสร้าง, การจัดลำดับขั้นตอนและการจัดวางองค์ประกอบของโปรแกรมคอมพิวเตอร์คือ คดี Whelan Associates V. Jaslow Dental Laboratories<sup>3</sup> ซึ่งคำตัดสินของศาลในคดีนี้ทำให้เกิดความเสียหายแก่วงการอุตสาหกรรมคอมพิวเตอร์เป็นอย่างมาก และถือว่าเป็นการบิดเบือนวัตถุประสงค์ของกฎหมายลิขสิทธิ์ ศาลในคดีอื่นที่ได้ยึดถือคำตัดสินของศาลในคดี whelan กับ Jaslow ด้วย เช่น ในคดี Pearl Systems Inc. V. Competition Electronics, Inc.<sup>4</sup> และคดี Lotus

<sup>1</sup> 462 F.Supp.1003(N.D. Tex. 1978)

<sup>2</sup> 605 F. Suppy. 816 819-21 ( M.D. Tenn. 1985 )

<sup>3</sup> 797 F.2d 1222,230 U.S.P.Q.481 (1986)

<sup>4</sup> 8 U.S.P.Q. 2d. 1420 ( S.D. Fla. 1988)

Development Corp. V. Pater Back Software Int.<sup>5</sup> จนกระทั่งได้มีการเปลี่ยนแนวคำตัดสินของศาลที่ได้สร้างบรรทัดฐานใหม่ในคดี Computer Associates International, Inc. V. Altai Inc.<sup>6</sup>

ก่อนที่ศาลจะมีคำตัดสินคดี whelan ได้มีคดี Synercom Technology V. University Computing Co., ซึ่งเป็นคดีที่เกิดขึ้นในช่วงแรก ๆ ที่เกี่ยวข้องกับคอมพิวเตอร์โปรแกรมและกฎหมายลิขสิทธิ์ ศาลพบว่ารูปแบบการรับข้อมูล(input formats) ไม่สามารถเปรียบเทียบกับงานวรรณกรรมอื่น ๆ ได้และเป็นเพียงความคิด (idea) ไม่ใช่เป็นการแสดงออกของความคิด ดังนั้นกฎหมายลิขสิทธิ์ไม่คุ้มครองรูปแบบการรับข้อมูล(input formats) ที่เป็นการจัดลำดับลงในโปรแกรมคอมพิวเตอร์ โดยทั่วไปผู้ใช้โปรแกรมจะถ่ายโอน (load) ข้อมูลลงในคอมพิวเตอร์ ตามการใส่ข้อมูล (input data) ของโปรแกรมคอมพิวเตอร์โดยการทำให้อำนาจเป็นบรรทัดต่อบรรทัด คำสั่งของโปรแกรมเปลี่ยนจากข้อมูลไปสู่ผลลัพธ์ที่ต้องการ หรือที่เรียกว่า Out put<sup>7</sup> ศาลเห็นว่า input format เป็นการถ่ายทอดข้อมูลเกี่ยวกับการจัด (sequencing) และลำดับ (ordering) ของข้อมูล เหตุผลของศาลที่ตัดสินว่า formats ไม่อยู่ภายใต้การคุ้มครองของกฎหมายลิขสิทธิ์เพราะว่า สิ่งเหล่านี้เป็นเพียงความคิดไม่ใช่การแสดงออกของความคิด ศาลยึดถือว่าโครงสร้าง (structure) และการจัดวางองค์ประกอบของการใส่ข้อมูลที่พื้นฐาน (simple input format) เป็นความคิดและไม่สามารถแยกเป็นการแสดงออกได้<sup>8</sup> ดังนั้นศาลโดยผู้พิพากษา แพทริก ฮิกกินส์เบอร์ธแฮม (Patrick Higginbotham) ตัดสินไม่ให้การคุ้มครองลิขสิทธิ์ในกรณีที่เป็นการจัดรูปแบบลำดับของข้อมูล (Input format) แม้ว่าบริษัทโจทก์ได้ไปยื่นขอจดทะเบียนลิขสิทธิ์ต่อสำนักงานลิขสิทธิ์แล้วก็ตาม เพราะการขอจดทะเบียนลิขสิทธิ์นั้น สำนักงานลิขสิทธิ์จะไม่พิจารณาในรายละเอียด แต่เป็นเพียงการจดทะเบียนเพื่อเป็นหลักฐานในเบื้องต้นเท่านั้นและถ้ามีข้อขัดแย้งประการใดศาลจะต้องเป็นผู้พิจารณาตัดสินชี้ขาด ซึ่งระบบนี้เป็นระบบที่ใช้ในประเทศสหรัฐอเมริกา สำหรับประเทศไทยไม่มีระบบการจดทะเบียนลิขสิทธิ์ตามแบบของสหรัฐอเมริกา จึงเห็นได้ว่าคำตัดสินของศาลในคดีดังกล่าวเป็นไปตามหลักของกฎหมายลิขสิทธิ์ที่ว่ากฎหมายลิขสิทธิ์นั้นไม่คุ้มครองไปถึงสิ่งที่เป็นความคิด แต่จะคุ้มครองการแสดงออกของความคิดเท่านั้น เพราะการที่ให้การคุ้มครองไปถึงความคิดจะเป็นการให้อำนาจ

<sup>5</sup> 740 F. Supp.37 (D. Mass 1990)

<sup>6</sup> 9820F.2d 693 ( 2d cir. 1992)

<sup>7</sup> Thomas M. Gage, “ Whelan Associates V. Jaslow Dental Laboratories : Copyright Protection for Computer Software Structure- What’s the purpose?,” Wisconsin Law Review 5, (May 1987): 870 , supra note 82.

<sup>8</sup> Ibid., :870.

ผูกขาดแก่เจ้าของลิขสิทธิ์ที่มีสิทธิ์แต่เพียงผู้เดียวในการใช้ความคิดนั้นซึ่งไม่ก่อให้เกิดประโยชน์แก่สังคมส่วนรวม

แต่ต่อมาได้มีคดี SAS Institute Inc. V. S&H Computer Systems Inc. ที่คำตัดสินของศาลได้ขยายขอบเขตการให้การคุ้มครองลิขสิทธิ์ในโปรแกรมคอมพิวเตอร์ให้กว้างมากขึ้น ซึ่งก่อให้เกิดสิทธิผูกขาดของเจ้าของลิขสิทธิ์ในการที่จะใช้ประโยชน์ในงานนั้นแต่เพียงผู้เดียว คดีนี้เป็นจุดเริ่มต้นของการบิดเบือนหลักของกฎหมายลิขสิทธิ์

คดีนี้มีข้อเท็จจริงว่า SAS Institute Inc. โจทก์ ได้เขียนโปรแกรมวิเคราะห์สถิติ (Statistical analysis program) ที่ใช้สำหรับเครื่องคอมพิวเตอร์ไอบีเอ็ม 370 (IBM System /370) ซึ่งเป็นเครื่องคอมพิวเตอร์ขนาดใหญ่ โปรแกรมของ SAS ได้รับความนิยมจากลูกค้ามาก มีลูกค้าจำนวนมากต้องการใช้โปรแกรมของ SAS แต่ไม่สามารถใช้โปรแกรมนี้ได้เนื่องจากไม่ได้ใช้เครื่องไอบีเอ็ม 370 จากการที่ SAS Institute ค้นพบว่าจากผลของการสำรวจตลาดคอมพิวเตอร์ปรากฏว่าลูกค้าต้องการให้โปรแกรมของ SAS นี้ปฏิบัติการหรือทำงานบนเครื่องคอมพิวเตอร์ VAX ที่ผลิตโดย Digital Equipment Corporation (DEC) ดังนั้น SAS จึงมีโครงการที่จะพัฒนาดัดแปลงโปรแกรมของ SAS เอง เพื่อให้ใช้กับเครื่องของ DEC ได้ แต่เนื่องจากธุรกิจเกี่ยวกับคอมพิวเตอร์นี้มีการแข่งขันกันสูงมาก จึงไม่ได้มีแต่เพียงบริษัท SAS เท่านั้นที่เห็นถึงโอกาสที่จะทำให้ลูกค้าของ DEC พึงพอใจกับโปรแกรมวิเคราะห์สถิติ ที่สามารถทำงานบนเครื่องของตนได้ S&H ได้ดำเนินการเขียนโปรแกรมขึ้นใหม่โดยใช้ภาษาในการเขียนโปรแกรมที่แตกต่างไปจากของ SAS และโปรแกรมใหม่มีชื่อว่าโปรแกรมอินดาส (INDAS) บริษัท SAS จึงได้ดำเนินการฟ้องร้องบริษัท S&H จำเลย ว่าได้ละเมิดลิขสิทธิ์ในโปรแกรมวิเคราะห์สถิติของโจทก์ และทำให้โจทก์สูญเสียผลประโยชน์เป็นจำนวนมากในอนาคตที่จะได้จากผู้ที่ใช้เครื่อง VAX

ในคดีนี้ผู้พิพากษา Thomas A. Wisemar Jr. ได้ปฏิเสธเหตุผลของ S&H ที่ว่าการกระทำของเขาเป็นไปตามวัตถุประสงค์ในการอนุญาตตามสัญญาอนุญาตให้ใช้สิทธิ (license) ของ SAS ที่ S&H ได้ใบอนุญาตให้ใช้สิทธิในฐานะเป็นลูกค้าธรรมดา (a standard customer license) สาเหตุที่ S&H ต้องดัดแปลงโปรแกรมของ SAS เพราะว่าโปรแกรมของ SAS ไม่สามารถใช้กับเครื่อง VAX ได้ เนื่องจากโปรแกรมของ SAS สร้างขึ้นมาเพื่อใช้งานบนเครื่องของ IBM ดังนั้นตัวแปลโปรแกรม(compiler) ของเครื่อง VAX จึงไม่สามารถแปลคำสั่งที่มีอยู่ในโปรแกรมคอมพิวเตอร์นั้นได้ จึงไม่สามารถเอาโปรแกรมของ SAS มาใช้กับเครื่อง VAX ได้

นอกจากนี้ในการพิจารณาศาลได้พบว่าโปรแกรมวิเคราะห์สถิติของโจทก์กับโปรแกรมอินดาซของจำเลย มีความเหมือนคล้ายกันในสาระสำคัญอย่างมากและพบหลักฐานต่าง ๆ ดังนี้<sup>9</sup>

1. ศาลพบว่าโปรแกรมย่อย 44 โปรแกรมของ S&H มีความเหมือนคล้ายกันกับโปรแกรมของ SAS (โจทก์) จนถือได้ว่าเป็นการคัดลอกโปรแกรมของโจทก์

2. ข้อเท็จจริงที่ว่า S&H ทำลายโปรแกรมต้นฉบับดั้งเดิม (the original source code) สำหรับผลิตภัณฑ์ของเขา ภายหลังจากที่ SAS ได้เห็นแล้วและได้ทำลายก่อนที่พยานผู้เชี่ยวชาญของ SAS จะสามารถตรวจสอบได้

3. ข้อเท็จจริงที่ว่ารหัสของ S&H ถูกเปลี่ยนโดยวิธีการที่เมื่อเปลี่ยนแล้วโปรแกรมคอมพิวเตอร์ไม่สามารถทำงานได้ (nonfunctional way) ก่อนที่ผู้เชี่ยวชาญของศาลจะได้พิจารณา เช่น โปรแกรมของ S&H พิมพ์คำว่า Statistical Analysis System ที่ส่วนต้นของ output ซึ่งข้อความนี้ต่อมาได้ถูกเปลี่ยนไป

4. ข้อเท็จจริงที่ว่าโปรแกรมรุ่นแรกของ S&H มีชื่อย่อ SAS ปรากฏอย่างน้อย 145 ข้อความที่แตกต่าง แต่สิ่งที่ปรากฏได้นี้ถูกขจัดออกไปในภายหลัง

5. เอกสารพื้นฐานของทำรหัส(coding) เผยแพร่ต่อโปรแกรมเมอร์บนโครงการ S&H ซึ่งชี้ว่าโครงสร้างโปรแกรมย่อย(modular) ของโปรแกรมมีแนวโน้มที่คัดลอก(duplicate) โปรแกรมของ SAS และชื่อของ "driver routines" ที่เป็นชื่อภายในที่ผู้ใช้โปรแกรมจะไม่สามารถเห็นได้ซึ่งถูกเขียนโดยโปรแกรมเมอร์ของ S&H เป็นสิ่งที่ใกล้เคียงกับในโปรแกรม SAS และนี่เป็นเพียงตัวอย่างเดียวที่โปรแกรมเมอร์ของ S&H ไม่ได้ใช้โปรแกรมย่อยของ SAS ที่เฉพาะนี้

6. การขาดเอกสารที่เป็นหลักฐานในการออกแบบโปรแกรม S&H เมื่อเปรียบเทียบกับหลักฐานต่าง ๆ ที่ใช้สำหรับโปรแกรม PASQUEC ที่ล้มเหลวเพราะไม่สมบูรณ์ (แสดงว่า PASQUEN ทำโดยบริษัท SAS แต่ล้มเหลว)

7. ข้อเท็จจริงที่ว่า การตรวจสอบ (testimony) เกี่ยวกับการอนุญาตให้ใช้สิทธิ (licensing) ศาลพบว่าสัญญาอนุญาตให้ใช้สิทธิไม่อนุญาตให้ทำการแยก(breaking)โปรแกรมคอมพิวเตอร์เพื่อเอาข้อมูลที่จำเป็นในการเขียนโปรแกรม การกระทำของ S&H ที่ไปเอาข้อมูลที่จำเป็นต่อโปรแกรมเมอร์ในการเขียนโปรแกรมจึงเป็นการกระทำที่ไม่ได้รับอนุญาต

จากข้อเท็จจริงที่ศาลพบดังกล่าว บริษัท S&H พยายามอ้างข้อต่อสู้ที่ว่า บริษัทได้ผลิตโปรแกรมนั้นขึ้นมาเพื่อพัฒนาโปรแกรมให้สามารถใช้งานกับเครื่องของ VAX ได้ และหลักพื้นฐานของกฎหมายลิขสิทธิ์ที่ว่ากฎหมายลิขสิทธิ์คุ้มครองการแสดงออกของความคิด แต่ไม่คุ้มครอง

<sup>9</sup> Anthony Lawrence Clapes, *Software Copyright and Competition : The look and feel of the law* , pp. 65-66.

ครองความคิดและโปรแกรมของ S&H สร้างขึ้นมาจากความคิด (idea) ของ SAS ซึ่งไม่ได้เป็นการแสดงออกของความคิดที่กฎหมายลิขสิทธิ์ให้การคุ้มครอง ซึ่งข้อต่อสู้ของจำเลยนี้เป็นสิ่งที่เกิดขึ้นในทุกคดีการละเมิดลิขสิทธิ์ ศาลสรุปว่าหลักฐานของ S&H เป็นเท็จข้อต่อสู้รับฟังไม่ได้

ขณะที่บริษัท S&H ผลิตโปรแกรมเพื่อให้เครื่องของ DEC ทำงานวิเคราะห์สถิติได้ เหมือนกับที่เครื่องของ IBM ทำงานเมื่อใช้โปรแกรมของ SAS ศาลเห็นว่าสามารถกระทำได้ ถ้าทำการสร้างสรรค์งานนั้นขึ้นมาด้วยตนเองไม่ได้ลอกเลียนผู้อื่น นอกจากนี้ศาลยังเห็นว่าการละเมิดลิขสิทธิ์ในโปรแกรมคอมพิวเตอร์นั้นอาจเป็นการละเมิดโดยปราศจากการเหมือนกันทางตัวอักษร เช่น การดัดแปลงโปรแกรมหนึ่งจากที่ใช้ในเครื่องคอมพิวเตอร์หนึ่งไปสู่เครื่องคอมพิวเตอร์อีกประเภทหนึ่ง โปรแกรมที่ดัดแปลงนั้นไม่มีการเหมือนกันทางตัวอักษรเพราะใช้ภาษาในการเขียนโปรแกรมที่แตกต่างกัน แต่กฎหมายลิขสิทธิ์คุ้มครองต่อต้านการดัดแปลงโดยไม่ได้รับอนุญาตนี้

S&H ได้ผลิตโปรแกรมคอมพิวเตอร์ที่คล้ายคลึงกับโปรแกรมของ SAS จนแทบจะกล่าวได้ว่า S&H ได้ลอกเลียนแบบการแสดงผลซึ่งความคิดของ SAS ซึ่งถือว่าการกระทำของจำเลยเป็นการละเมิดลิขสิทธิ์โปรแกรมคอมพิวเตอร์ของโจทก์

การพิสูจน์ถึงการกระทำที่เป็นการละเมิดลิขสิทธิ์นั้น ความเหมือนคล้ายกันในส่วนที่เป็นสาระสำคัญ (substantial similarity) เป็นสิ่งที่มีความจำเป็นมากเพราะถ้าโปรแกรมของโจทก์และจำเลยมีความเหมือนคล้ายกันในส่วนที่เป็นสาระสำคัญ ก็ถือว่าจำเลยได้ละเมิดลิขสิทธิ์ในโปรแกรมคอมพิวเตอร์ของโจทก์ ศาลในคดีนี้ได้พิจารณาถึงความเหมือนคล้ายกันในส่วนที่เป็นสาระสำคัญ และได้มีเหตุผลมากมายในการปฏิเสธข้อแก้ตัวหรือข้อต่อสู้ของจำเลย 4 ประการคือ<sup>10</sup>

1. การทดสอบความเหมือนคล้ายในสาระสำคัญเป็นเรื่องเกี่ยวกับคุณสมบัติ (คุณภาพ) ไม่ใช่เป็นเรื่องเกี่ยวกับปริมาณส่วนที่เล็กแต่เป็นส่วนที่มีความสำคัญหรือที่มีคุณภาพของงานที่ได้รับการคุ้มครอง อาจถือว่าเป็นส่วนอันเป็นสาระสำคัญของโปรแกรมคอมพิวเตอร์ เพราะว่า ส่วนอันเป็นสาระสำคัญไม่มีขึ้นอยู่กัปริมาณ

2. ตัวอย่าง 44 โปรแกรมย่อยที่ได้ถูกลอกเลียนมาไม่สามารถกล่าวได้ว่าเป็นจำนวนที่ไม่สำคัญ โปรแกรมย่อยบนโปรแกรมที่เหมือนระหว่างโปรแกรมของโจทก์และของจำเลยนี้ ทำให้สรุปได้ว่าการลอกเลียนเกิดขึ้น

<sup>10</sup> Ibid., p. 67.



3. ถ้า S&H ไม่ทำลายหรือปิดบังหลักฐานต่าง ๆ ของการคัดลอก (copying) ก็จะมีตัวอย่างที่มากกว่านี้

4. หลักฐานของการทำซ้ำแสดงให้เห็นว่าไม่เพียงบางส่วนของรหัสที่เหมือนกัน แต่การจัดวางองค์ประกอบ (organization) และโครงสร้าง (structure) ของโปรแกรม SAS ก็มีหลักฐานแสดงให้เห็นว่ามีการกระทำการลอกเลียนโดย S&H แม้ว่าโปรแกรม S&H เป็นไปได้ว่าได้บรรจุสิ่งสำคัญของงานดั้งเดิม (original work) ศาลสรุปว่ามันเป็นส่วนสาระสำคัญและโปรแกรมของจำเลยมีรากฐานมาจากโปรแกรมของโจทก์

เพื่อจะได้เป็นประโยชน์ในการทำความเข้าใจถึงคดีมากยิ่งขึ้นผู้เขียนจะกล่าวถึงความหมายของโปรแกรมประกอบหรือโปรแกรมย่อยที่มีความหมายอย่างไร โปรแกรมประกอบ (Subprogram) หมายถึงโปรแกรมที่ต้องใช้ประกอบกับโปรแกรมอื่นๆ นั่นคือ โปรแกรมประกอบไม่สามารถทำงานได้โดยลำพัง ในการเขียนโปรแกรมนั้นเราพบว่าบางโปรแกรมมีช่วงการทำงานที่ซ้ำกัน ต่างกันเฉพาะชื่อตัวแปรที่ใช้สำหรับแต่ละช่วง ดังนั้นเพื่อเป็นการประหยัดเวลาในการเขียนโปรแกรม การเจาะบัตร การทดสอบเพื่อแก้ไข และเนื้อที่ที่ใช้เก็บช่วงคำสั่งที่ซ้ำๆ กันนั้น เราจึงเขียนชุดคำสั่งสำหรับทำงานที่ซ้ำๆ กันนี้แยกออกมาต่างหากจากโปรแกรมหลัก (Main Program) และเรียกโปรแกรกดังกล่าวว่าโปรแกรมประกอบ (Subprogram)<sup>11</sup> ในบางครั้งอาจเรียกโปรแกรมประกอบว่า โปรแกรมย่อย ก็ได้ซึ่งมีความหมายเหมือนกัน

นอกจากนี้ศาลยังเห็นว่าประชาชนควรถูกห้ามสำหรับการแปลโปรแกรมคอมพิวเตอร์อื่นๆ และอ้างว่าสิ่งที่แปลนั้นเป็นของตนเอง การกระทำอย่างนี้ถือว่าเป็นการละเมิดลิขสิทธิ์ในโปรแกรมคอมพิวเตอร์

ในคดี SAS ศาลได้ขยายการคุ้มครองโดยให้การคุ้มครองไปถึงการจัดวางองค์ประกอบและโครงสร้างของโปรแกรมคอมพิวเตอร์ด้วย แต่คดีที่เป็นบรรทัดฐานที่ศาลให้การคุ้มครองแก่การออกแบบโครงสร้าง (structure), การจัดลำดับขั้นตอน (sequence) และการจัดวางองค์ประกอบ (organization) ของโปรแกรมคอมพิวเตอร์คือ คดี Whelan Associates, Inc. V. Jaslow Dental Laboratories<sup>12</sup>

<sup>11</sup> ศิริพร สาเกตทอง, ความรู้เบื้องต้นเกี่ยวกับคอมพิวเตอร์และหลักภาษาฟอร์เทน IV, หน้า 263.

<sup>12</sup> 797 F. 2d 1222, 230 U.S.P.Q. 481(1986) code in, Anthony Lawrence Clapes, Software Copyright and Competition : The look and feel of the law, pp. 94-109.

คดีที่บริษัท Whelan Associate, Inc. เป็นโจทก์ฟ้องว่าบริษัท Jaslow Dental Laboratories, Inc. ละเมิดลิขสิทธิ์ในโปรแกรมคอมพิวเตอร์ของโจทก์ ซึ่งข้อเท็จจริงมีว่า บริษัท Jaslow เป็นบริษัทที่ให้บริการอำนวยความสะดวกแก่คลินิกทันตแพทย์ ในการทำรายการสินค้า, การสั่งซื้อสินค้าเกี่ยวกับอุปกรณ์ทำฟัน, การทำรายการสั่งซื้อสินค้าของลูกค้า, การส่งสินค้า ซึ่งข้อมูลต่างๆ เหล่านี้ต้องมีการเปลี่ยนแปลงตลอดเวลา และเนื่องจากอุปกรณ์ทางทันตกรรมมีรายละเอียดมากและมีราคาสูง บริษัท Jaslow โดยนาย Rand Jaslow ได้หาทางที่จะบริหารงานของบริษัท Jaslow ใหม่โดยพยายามที่จะใช้เครื่องคอมพิวเตอร์เข้ามาช่วยในการทำงานเพราะการใช้เครื่องคอมพิวเตอร์นี้จะทำให้การทำงานสะดวกรวดเร็วมากขึ้น และมีความถูกต้องแม่นยำมากกว่าเดิม แต่เนื่องจากตัวเขาเองไม่มีความรู้ทางด้านคอมพิวเตอร์ เขาจึงทำสัญญาว่าจ้างให้บริษัท Strohl Systems เป็นผู้ดำเนินการเขียนโปรแกรมคอมพิวเตอร์ให้โดยบริษัท Strohl Systems มีนาง Elain Whelan ซึ่งเป็นผู้ถือหุ้นของบริษัทนี้เป็นหัวหน้าทีมในการออกแบบและเขียนโปรแกรม โปรแกรมที่ถูกเขียนนี้มีชื่อว่า Dentalab เขียนสำเร็จเมื่อปี 1979 โดยเขียนในภาษา EDL (Event Driven Language) และให้ใช้งานบนเครื่อง IBM Series I

สัญญาว่าจ้างที่บริษัท Jaslow ว่าจ้างให้บริษัท Strohl Systems เขียนโปรแกรมนี้ระบุว่า โปรแกรม Dentalab เป็นลิขสิทธิ์ของบริษัท Strohl Systems ซึ่งต่อมานาง Elain Whelan ได้ซื้อหุ้นของบริษัทนี้จากหุ้นส่วนคนอื่นแล้วเปลี่ยนชื่อเป็นบริษัท Whelan Associates, Inc. สัญญานี้ก็ตกไปเป็นของบริษัท Whelan ตามหลักกฎหมายเรื่องสัญญา ดังนั้นเจ้าของลิขสิทธิ์ในโปรแกรมคอมพิวเตอร์ Dentalab นี้เป็นของบริษัท Whelan แต่บริษัท Whelan ได้มอบหมายให้บริษัท Jaslow เป็นตัวแทนในการจำหน่ายโปรแกรมนี้ให้แก่คลินิกทันตแพทย์ที่สนใจในโปรแกรมนี้

ในช่วงเวลานั้นตลาดของเครื่องคอมพิวเตอร์มีการเปลี่ยนแปลงโดยเครื่องคอมพิวเตอร์ส่วนบุคคล (Personal Computer หรือ PC) ได้รับความนิยมมากขึ้นเพราะเนื่องจากมีราคาถูกกว่าเครื่องคอมพิวเตอร์ขนาดใหญ่และเหมาะสมกับกิจการขนาดเล็กที่ใช้การลงทุนที่ไม่สูงมาก แต่เนื่องจากโปรแกรม Dentalab ไม่สามารถนำไปใช้กับเครื่อง IBM PC ได้ นาย Rand Jaslow ได้สังเกตเห็นถึงปัญหานี้จึงได้ปรึกษากับโปรแกรมเมอร์คือ นาย Jonathan Novak เพื่อเขียนโปรแกรมคอมพิวเตอร์ใหม่โดยใช้ภาษา BASIC ที่สามารถใช้กับเครื่องคอมพิวเตอร์ส่วนบุคคลได้ โปรแกรมใหม่นี้มีชื่อว่าโปรแกรม Dent-com Program

บริษัท Whelan ได้ยื่นฟ้องบริษัท Jaslow ต่อศาล United States District Court for the Eastern District of Pennsylvania โดยกล่าวหาว่าโปรแกรมคอมพิวเตอร์ Dent-com Program ของบริษัทจำเลยละเมิดลิขสิทธิ์ในโปรแกรมคอมพิวเตอร์ Dentalab ของบริษัท Whelan (โจทก์) โดยมีผู้พิพากษา โด널ด์ แวน อาร์ตสดาเลน (Donald Van Artsdalen) เป็นผู้ตัดสินในคดีนี้

จากข้อเท็จจริงพบว่าการเขียนโปรแกรม Dent-Com Program นาย Rand Jaslow ได้มีการเข้าถึงและใช้โปรแกรมต้นฉบับ(source code) ของโปรแกรม Dentalab ของบริษัท Whelan ซึ่งเป็นการใช้ประโยชน์ในงานต้นฉบับอันมีลิขสิทธิ์ในการเตรียมงานอื่น ๆ ที่มีมาภายหลังและการทำงานใหม่และงานเดิมมีความเหมือนคล้ายกันในสาระสำคัญก็เพียงพอที่จะเป็นหลักฐานของการละเมิดลิขสิทธิ์<sup>13</sup> นอกจากนั้นนาย Rand Jaslow ได้โปรแกรมต้นฉบับ(source code) มาโดยไม่ได้รับอนุญาตจากบริษัท Whelan ในการเขียนโปรแกรม Dent-com Program และได้มีการลอกเลียนในส่วนของโครงสร้างของข้อมูล (data structures), ผังงาน (flows), รูปแบบหน้าจอ(screen formats), ภาษา(language), รัสพจน์(acronyms)\*, ระบบการจัดเรียง(collating methods) และหน้าที่(functions) ทั้งหมดของโปรแกรม Dentalab

บริษัท Jaslow จำเลยได้อ้างข้อต่อสู้ว่าลิขสิทธิ์ในโปรแกรม Dentalab เป็นของตน ไม่ได้เป็นของโจทก์เพราะว่า Rand Jaslow เป็นผู้สร้างสรรค์ร่วมของโปรแกรมนี้แต่ไม่มีรายชื่อในแบบฟอร์มของการจดทะเบียนลิขสิทธิ์ (Copyright registration form) และโดยที่บริษัท Jaslow ได้ว่าจ้างให้ Whelan สร้างโปรแกรม Dentalab ดังนั้นบริษัท Jaslow จึงเป็นเจ้าของลิขสิทธิ์ในโปรแกรมนี้ นอกจากนี้ยังมีข้อต่อสู้อีกว่าโปรแกรม Dent-Com Program ถูกสร้างโดยอิสระโดยนาย Rand Jaslow และกฎหมายลิขสิทธิ์ไม่คุ้มครองเจ้าของลิขสิทธิ์ในการที่จะขัดขวางการที่ใครคนอื่นที่จะผลิตงานที่เหมือนคล้ายกับงานตน โดยที่เขาได้สร้างสรรค์งานโดยอิสระไม่ได้มีการคัดลอกของใครมา การกระทำของบริษัท Whelan ถือเป็นการจำกัดความเป็นอิสระในการแข่งขัน (unfair competition) เพราะถ้าบริษัท Whelan สามารถที่จะห้ามคนอื่นไม่ให้ทำงานที่เหมือนคล้ายกับงานตนโดยที่คนอื่นได้สร้างสรรค์งานนั้นขึ้นมาเองแล้วการกระทำดังกล่าวถือว่าต้องเป็นการแข่งขันที่ไม่เป็นธรรมซึ่งผิดกฎหมาย

นาย Rand Jaslow ได้พิสูจน์เพื่อแสดงให้เห็นว่าตัวเองเป็นผู้สร้างสรรค์ร่วมด้วยโดยพิสูจน์ว่า ความคิดเริ่มแรกและการเตรียมงาน (ผลงาน) ของ Dental และ Laboratory เป็นของ

<sup>13</sup> Ibid., p. 97.

\* รัสพจน์ (acronym) หมายถึง คำย่อที่เกิดจากการนำอักษรตัวแรก หรือกลุ่มอักษรบางตัวมาสร้างขึ้นเป็นคำใหม่ประกอบด้วยอักขระตั้งแต่สองตัวขึ้นไป นิยมออกเสียงเหมือนคำปกติ เช่น DOS ซึ่งเอาอักษร D มาจากคำว่า Disk อักษร O มาจาก คำ Operating และ S มาจาก System คำนี้อ่านออกเสียงว่า “ดอส” แทนที่จะอ่านว่า “ดีโอเอส”

ทักษิณา สวานานนท์, ศัพท์คอมพิวเตอร์สำหรับผู้ปฏิบัติงานแมคอินทอชและพีซี, (กรุงเทพฯ: ห้างหุ้นส่วนจำกัด วี.ที.ซี. คอมมิวนิเคชั่น, 2538), หน้า 3.

ตนและตนเป็นคนเปิดเผยให้ Elaine Whelan รู้ถึงรายละเอียดและวิธีการดำเนินธุรกิจของบริษัท Jaslow Laboratories และเป็นคนอธิบายถึงหน้าที่ต่าง ๆ ที่จะถูกปฏิบัติการโดยโปรแกรม อีกทั้งยังช่วยออกแบบภาษาและรูปแบบการแสดงผลทางหน้าจอของโปรแกรม อย่างไรก็ตามศาลเห็นว่า Elaine Whelan แต่เพียงผู้เดียวเป็นผู้สร้างสรรค์โปรแกรม Dentalab เพราะโปรแกรม Dentalab เกิดจากการใช้ความชำนาญและการสร้างสรรค์ของ Whelan ในการสร้างโครงสร้างข้อมูล (Data structures), ผังงาน (Work flows), จอภาพ (Screens) และการติดต่อกันระหว่างส่วนต่าง ๆ ของโปรแกรมที่ถูกออกแบบ

สาระสำคัญของคดีนี้ไม่ใช่เรื่องการลอกเลียนตัวอักษร (Non-Verbal expression) หรือ คัดลอกตัวอักษรโดยตรง (Literal copy) เพราะโปรแกรมของโจทก์ใช้ภาษา EDL ส่วนโปรแกรมของจำเลยใช้ภาษา BASIC เป็นการใช้ภาษาที่แตกต่างกันจึงไม่มีการคัดลอกตัวอักษร แต่คดีนี้จะเป็นเรื่องของการคุ้มครองเค้าโครง (structures), การจัดลำดับตอน (sequence) และการจัดวางองค์ประกอบ (organization) ของโปรแกรมคอมพิวเตอร์ ดังนั้นในการพิสูจน์ถึงการละเมิด ลิขสิทธิ์ศาลต้องหาหลักการในการพิจารณา ซึ่งต่างกับอย่างคดี Apple Computer V. Franklin Computer ที่เป็นการทำซ้ำในโปรแกรมภาษาเครื่อง (object code) และเป็นการคัดลอกตัวอักษร ดังนั้นหลังจากที่ได้พิจารณาแล้วว่าใครเป็นผู้ที่มีลิขสิทธิ์ในงานอันนี้ศาลในคดี Whelan V. Jaslow จึงได้วางแนวทางในการพิสูจน์ 2 วัชขั้นตอนคือ

1. ศาลต้องพิสูจน์ว่าจำเลยได้มีโอกาสเข้าถึงและใช้ประโยชน์จากโปรแกรมต้นฉบับ (source code) ของโจทก์หรือไม่ ขั้นตอนนี้จะเป็นการพิสูจน์ถึงการกระทำของจำเลยที่เข้าถึงงานโปรแกรมคอมพิวเตอร์ของโจทก์เพราะถ้าจำเลยสร้างสร้งงานนั้นขึ้นมาเอง โดยมีได้ลอกเลียนแบบใคร ถึงแม้งานที่ออกมาจะเหมือนกันจำเลยก็มีลิขสิทธิ์ในงานนั้นและไม่ได้ถือว่าเป็นการละเมิดลิขสิทธิ์ในงานของโจทก์ เนื่องจากกฎหมายลิขสิทธิ์ไม่ยึดหลักความใหม่ (novelty) ของงาน เพราะถ้าจำเลยได้เข้าถึงงานของโจทก์และงานของจำเลยมีความเหมือนคล้ายกับงานของโจทก์นี้การกระทำของจำเลยมีแนวโน้มที่จะเป็นการละเมิดลิขสิทธิ์ในโปรแกรมคอมพิวเตอร์ของโจทก์

2. ศาลต้องพิสูจน์ว่าโปรแกรม Dent-com ของจำเลยนั้นมีความเหมือนคล้ายกันในส่วนอันเป็นสาระสำคัญกับโปรแกรม Dentalab ของโจทก์หรือไม่ ขั้นตอนนี้เป็นขั้นตอนของการพิสูจน์ความเหมือนคล้ายในส่วนอันเป็นสาระสำคัญ (substantial similarity) เพราะถ้าโปรแกรมของจำเลยไม่มีความเหมือนคล้ายในสาระสำคัญกับโปรแกรมของโจทก์ จำเลยก็ไม่มีคามผิดฐานละเมิดลิขสิทธิ์และโปรแกรมที่จำเลยเขียนนั้นถือว่าเป็นงานอันมีลิขสิทธิ์ได้ เพราะเกิดจากการสร้างสรรค์งานนั้นด้วยตนเอง

เนื่องจากในคดีนี้จำเลยได้มีโอกาสศึกษาถึงโปรแกรมต้นฉบับ(source code) ของโปรแกรมของโจทก์ตามที่ได้กล่าวมาแล้วข้างต้น ดังนั้นจึงผ่านการพิสูจน์ในขั้นตอนที่หนึ่งแล้ว ต่อมาจะเป็นขั้นตอนที่สำคัญที่สุดในการพิจารณาว่าโปรแกรมของจำเลยละเมิดลิขสิทธิ์ในโปรแกรมคอมพิวเตอร์ของโจทก์หรือไม่ นั่นคือ การพิจารณาว่าโปรแกรม Dent-com ของจำเลยมีความเหมือนคล้ายในสาระสำคัญกับโปรแกรม Dentalab ของโจทก์หรือไม่

การพิสูจน์ว่างานทั้งสองนั้นมีความเหมือนคล้ายกันในสาระสำคัญหรือไม่นั้นมีแนวทางการพิจารณาในคดี Arnstein V. Porter<sup>14</sup> ซึ่งใช้แนวทางในการตรวจสอบความเหมือนคล้าย 2 ระดับ คือ

1.การตรวจสอบโดยผู้เชี่ยวชาญ(Expert Testimony) โดยมุ่งพิจารณาถึงรายละเอียดภายในของงานทั้งสองชิ้น (Intrinsic test of substantial similarity) ว่ามีความเหมือนคล้ายกันในสาระสำคัญหรือไม่

2.การตรวจสอบโดยบุคคลทั่วไป ที่มุ่งพิจารณาถึงความเหมือนคล้ายในสาระสำคัญจากภายนอก (Extrinsin test of substantial similarity)

เนื่องจากศาลในคดี Whelan ได้ปฏิเสธที่จะใช้การพิสูจน์โดยบุคคลทั่วไปเพราะลักษณะของโปรแกรมคอมพิวเตอร์เป็นงานที่มีความสลับซับซ้อนแตกต่างไปจากงานทั่วไป<sup>15</sup> ซึ่งหากใช้การพิจารณาของบุคคลทั่วไปที่ไม่มีความรู้เชี่ยวชาญก็อาจก่อให้เกิดข้อผิดพลาดได้ นอกจากนั้นศาลเห็นว่าข้อมูล, การใช้ถ้อยคำ (terminology) และการใช้อักษรย่อ (use of abbreviations) ของโปรแกรมของจำเลยส่วนใหญ่เหมือนกันทุกประการกับการแสดงทางหน้าจอ (screen display) ของโปรแกรม Dentalab ของโจทก์ โดยศาลสังเกตว่า ลูกคำต่างๆไปที่จะชื่อโปรแกรมกล่าวว่าไม่มีความแตกต่างในสาระสำคัญระหว่างโปรแกรม Dentalab และโปรแกรม Dent-com<sup>16</sup> ดังนั้นศาลจึงเห็นว่าการทดสอบโดยผู้เชี่ยวชาญมีความน่าเชื่อถือมากกว่าเพราะเป็นผู้ที่มีความรู้เชี่ยวชาญทางด้านโปรแกรมคอมพิวเตอร์สามารถที่จะพิจารณาถึงความเหมือนคล้ายในสาระสำคัญได้ดีกว่าบุคคลธรรมดาทั่วไปทั้งนี้เพราะโปรแกรมคอมพิวเตอร์มีลักษณะที่แตกต่างไปจากงานอันมีลิขสิทธิ์อื่นๆ เช่น นวนิยายหรือบทเพลงที่ง่ายต่อการพิสูจน์ถึงการละเมิดลิขสิทธิ์

<sup>14</sup> 154 F.2d 464 ( 2 nd Cir.1946)

<sup>15</sup> Thomas M. Gage, " Whelan Associates V. Jaslow Dental Laboratories : Copyright Protection for Computer Software Structure- What's the purpose?,": 878.

<sup>16</sup> Anthony Lawrence Clapes, Software Copyright and Competition : The look and feel of the law, p. 100.

การทดสอบความเหมือนคล้ายในสาระสำคัญของคดี Whelan ได้แบ่งออกเป็นสองขั้นตอนเหมือนคดี Arnstein ซึ่งขั้นตอนแรกเหมือนกับ Arnstein test ทุกประการคือพิจารณาว่าจำเลยได้ใช้งานของโจทก์ในการสร้างสรรค์งานที่ถูกกล่าวหาว่าจะเมิดลิขสิทธิ์หรือไม่ แต่ที่ต่างจากคดี Arnstein คือการทดสอบในขั้นตอนที่สองที่พิจารณาว่าโปรแกรมคอมพิวเตอร์ของโจทก์กับจำเลยมีความเหมือนคล้ายกันในสาระสำคัญหรือไม่ ศาลในคดี Whelan ได้พิจารณาถึงความไม่เหมาะสมของการพิสูจน์โดยบุคคลธรรมดา (Ordinary observer test) สำหรับการปรับใช้แก่คดีที่เกี่ยวข้องกับโปรแกรมคอมพิวเตอร์ ศาลไม่ต้องการขยายการทดสอบความเหมือนคล้ายในสาระสำคัญเพื่อขยายขอบเขตของการคุ้มครองลิขสิทธิ์ออกไป

ในคดี Whelan นี้ผู้เชี่ยวชาญพบว่าโปรแกรมของโจทก์และจำเลยมีความเหมือนคล้ายกัน ดังต่อไปนี้

1. หลักฐานแรกคือความเหมือนคล้ายกันในโครงสร้าง (structure), ฝั่ง (flow), ตรรกะ (logic) และการรวมชื่อ (naming conventions) ใน ลำดับการเข้าส่วนต่างๆ ทั้งหมดของทั้งสองโปรแกรม ดังต่อไปนี้<sup>17</sup>

#### Order Entry Programs

##### Dentalab System

Primary Menu, choose[1]  
Production SCHEDULING

Product Menu, choose[1]  
ORDER ENTRY  
DL1000 program(Order Entry)

“ENTRY ACCOUNT OR NAMEKEY  
“Check CUSTMAST for valid Customer. Read CUSTMAST file on this customer.

##### Dentcom PC Systems

Primary Menu, choose[1]  
Production MENU

Product Menu, choose[1]  
ORDER ENTRY  
ORDER ENTRY program

“ENTRY ACCOUNT NAME KEY:  
-----  
Check CUSTMAST for valid customer. If valid, read

<sup>17</sup> Idib., pp. 106-107.

Increment sequential order no.  
In ORDERS. Display customer  
name, address.

Display entry screen(6.6).  
patient shade, remake, call

Pan #. Dr.'s. request date.

Ask for first department number.  
Display dept. order screen(6.8-  
6.11)(list of items in this dept.  
from ITEMMAST)

User entry choices

System adds days in dept.[DAYVAL]  
to present date to find due out  
date. System computers work load  
for dept/day out by product of  
load factor x quantity

ITEMMAST

Adds this to load already in DAYVAL  
for date out

CUSTMAST file for this customer.

If yes, increment order# in  
ORDERS

Display entry screen, patient  
shade, mould, remake, call Dr.?  
Dr.?

Case/pan#. Drs. request date,  
final case status T.F.B.R.

IS THIS SCREEN CORRECT ?

If yes ask for first department  
number. Display dept. order  
screen(P 10)(list of item in  
this dept from ITEMMAST

User entry choices

System adds days in dept.  
from COMPANY to present date  
to find due out date. Time  
is of AM,"noon" or PM.  
System accumulates case load  
by product of item load  
factor x quantity

ITEMMAST

Adds this to load already  
In DEPTLOAD for date out.

2.หลักฐานที่ได้จากบางส่วนของ การทดสอบของผู้เชี่ยวชาญของ Whelan Associates คือ Dr. Thomas Moore เกี่ยวกับการเชื่อมต่อกันของ 2 โปรแกรมย่อยที่พบว่าใน

ทั้งสองโปรแกรมย่อยนั้นต่างก็ใช้ the invoicing subroutine และ month-end subroutine ที่เหมือนกัน

หลังจากที่ ศาลได้พิจารณาแล้วว่าโปรแกรมของโจทก์และจำเลยมีความเหมือนคล้ายกันในสาระสำคัญศาลต้องพิจารณาต่อไปอีกว่าโครงสร้าง, การจัดลำดับขั้นตอนและการจัดวางองค์ประกอบของโปรแกรมคอมพิวเตอร์เป็นสิ่งที่กฎหมายลิขสิทธิ์ให้การคุ้มครองหรือไม่ โดยศาลมีวิธีแยกความแตกต่างระหว่างความคิดและการแสดงออกของความคิด และศาลเห็นว่า การเขียนโปรแกรมสำหรับใช้ในการบริหารงานคลินิกทันตแพทย์นั้นเป็นเพียงแต่ความคิดเท่านั้น ดังนั้นบุคคลอื่นสามารถเขียนโปรแกรมเพื่อบริหารงานคลินิกได้ซึ่งวิธีการบริหารงานคลินิกนั้นมีหลายแบบและวิธีการเขียนโปรแกรมคอมพิวเตอร์นั้นเขียนได้หลายทาง ในเมื่อมีวิธีการหลากหลายในการเขียนโปรแกรมนี้ศาลจึงเห็นว่าสิ่งนี้เป็นการแสดงออกของความคิดที่กฎหมายลิขสิทธิ์ให้การคุ้มครอง (Plurality of expression test คือวิธีการทดสอบความหลากหลายของวิธีในการแสดงความคิด)

ดังนั้นศาลในคดี Whelan ได้ขยายการคุ้มครองแก่โปรแกรมคอมพิวเตอร์ให้ครอบคลุมไปถึงการออกแบบโครงสร้าง, การจัดลำดับขั้นตอนและรูปแบบองค์ประกอบของโปรแกรม คำตัดสินดังกล่าวถือได้ว่าเป็นข้อผิดพลาดอย่างมากของศาลสหรัฐอเมริกาเพราะเท่ากับว่าเป็นการก่อให้เกิดอำนาจผูกขาดแก่เจ้าของลิขสิทธิ์ในการใช้ประโยชน์จากงานนั้นแต่เพียงผู้เดียว และยังเป็น การกีดกันทางการค้าที่ก่อให้เกิดการแข่งขันที่ไม่เป็นธรรมได้ ซึ่งสิ่งเหล่านี้ไม่ใช่เจตนารมณ์ของกฎหมายลิขสิทธิ์

นอกจากนั้นยังมีคดี Pearl System V. Competition Electronics, Inc.,<sup>18</sup> ที่ได้ขยายขอบเขตการคุ้มครองแก่โครงสร้างของโปรแกรมมากกว่าคดี Whelan โดยในคดีนี้ศาลเห็นว่าโปรแกรมมีความเหมือนคล้ายกันจากการใช้ชุดคำสั่ง(module) 2 กลุ่มที่มีหน้าที่และมีโครงสร้างระดับสูงที่เหมือนกัน(similar high-level structure) ศาลเห็นว่าโครงสร้างของ 2 ส่วนของกลุ่มของชุดคำสั่งที่แทบจะเหมือนกันทุกประการศาลถือได้ว่าเป็นการกระทำที่ละเมิดลิขสิทธิ์โปรแกรมคอมพิวเตอร์ของโจทก์แล้ว<sup>19</sup> ศาลในคดีนี้ได้ขยายกฎหมายของคดี Whelan ออกไปเพราะในคดี Whelan มีความเหมือนคล้ายปรากฏในระดับโครงสร้างทั้งหมดของโปรแกรมคอมพิวเตอร์

<sup>18</sup> 8 U.S.P. Q. 2d(BNA) 1520 (S.D. Fla. 1988)

<sup>19</sup> David D. Meyer and Gregory B. Heller, "Idea, Process, or protected Expression? : Determining the Scope of Copyright Protection of the Structure of Computer Program," *Michigan Law Review* 88, No.4 (February 1990): 885.



แต่คดีนี้ศาลพบว่าลิขสิทธิ์ในโปรแกรมถูกละเมิดเพียง 2 กลุ่มชุดคำสั่งที่มีหน้าที่เหมือนกัน ในโครงสร้างระดับสูงและมีความเหมือนคล้ายทางพื้นฐานของการติดต่อเชื่อมประสานระหว่างผู้ใช้กับโปรแกรมคอมพิวเตอร์ (user interface) เท่านั้น

หลังจากที่ศาลของสหรัฐได้หลงทางไปกับการให้การคุ้มครองแก่โปรแกรมคอมพิวเตอร์ที่มากเกินไป ต่อมาศาลได้เปลี่ยนแนวคำพิพากษาไปโดยไม่ขยายการคุ้มครองไปถึงสิ่งที่เป็นโครงสร้างของโปรแกรมเพราะการให้การคุ้มครองไปถึงสิ่งนี้จะเป็นการให้สิทธิผูกขาดแก่เจ้าของโปรแกรมคอมพิวเตอร์มากเกินไปและยังเป็นการให้การคุ้มครองไปถึงความคิดที่เป็นการขัดกับหลักของกฎหมายลิขสิทธิ์ซึ่งจะไม่คุ้มครองไปถึงสิ่งที่เป็นเพียงความคิดเท่านั้น คดีที่เป็นแนวบรรทัดฐานอันใหม่คือ คดี Computer Associates International, Inc. V. Altai Inc.,

ในคดีนี้โจทก์คือบริษัท Computer Associates ได้ฟ้องบริษัท Altai ว่าโปรแกรม OSCAR 3.4 และโปรแกรม OSCAR 3.5 ของจำเลยละเมิดลิขสิทธิ์ในโปรแกรม ADAPTER ของโจทก์ ข้อเท็จจริงในคดีนี้ พบว่าโปรแกรม OSCAR 3.4 ของจำเลยได้เขียนขึ้นโดยนาย Arney O. เดิมเป็นผู้หนึ่งที่ได้ร่วมเขียนโปรแกรม ADAPTER ของโจทก์ แต่ต่อมาได้ลาออกจากบริษัทของโจทก์แล้วมาทำงานให้กับบริษัท Altai โดยที่นาย Arney ได้ลักลอบเอาโปรแกรมต้นฉบับ (source code) ของบริษัทโจทก์ออกมาและได้ใช้ประโยชน์ในการเขียนโปรแกรม OSCAR 3.4 โดยได้คัดลอกโปรแกรมดังกล่าวประมาณ 30 % ของโปรแกรมทั้งหมด ดังนั้นจึงเป็นที่ชัดเจนว่าโปรแกรม OSCAR 3.4 ของจำเลยละเมิดลิขสิทธิ์ในโปรแกรมคอมพิวเตอร์ของโจทก์

ส่วนที่มีปัญหาคือโปรแกรม OSCAR 3.5 เพราะโปรแกรมดังกล่าวเป็นโปรแกรมที่ได้เขียนขึ้นมาใหม่เพื่อขจัดส่วนที่ได้ลักลอบจากโปรแกรมของโจทก์มาประมาณ 30% โดยในการเขียนขึ้นมาใหม่นี้ได้ใช้วิธีการ Clear room และนาย Arney ไม่ได้เข้ามามีส่วนร่วมด้วยในการเขียนโปรแกรมนี้ ในคดีนี้ผู้พิพากษา Pratt ได้แต่งตั้งให้ Dr. Rondall Davis แห่งสถาบัน MIT (The Massachusetts Institute of Technology) เป็นผู้เชี่ยวชาญเพื่อให้คำปรึกษาแก่ผู้พิพากษา เพราะเนื่องจากคดีนี้เกี่ยวข้องกับเทคโนโลยีทางด้านโปรแกรมคอมพิวเตอร์ที่มีความสลับซับซ้อนจำเป็นที่ศาลจะต้องมีผู้เชี่ยวชาญเป็นที่ปรึกษาในการตัดสินคดี และนอกจากนั้นคดีนี้เป็นคดีที่ไม่มีการพิจารณาโดยลูกขุนเนื่องจากศาลเห็นว่าเป็นคดีที่เกี่ยวข้องกับเทคโนโลยีคอมพิวเตอร์ที่ต้องพิจารณาจากหลักฐานทางเทคโนโลยีเป็นสำคัญและคู่กรณีทั้งสองฝ่ายได้แต่งตั้งผู้เชี่ยวชาญเข้ามาเป็นผู้มีส่วนร่วมในการดำเนินขบวนการพิจารณาแล้ว

ผู้พิพากษา Pratt ใช้วิธีพิจารณาถึงความเหมือนคล้ายระหว่างโปรแกรมคอมพิวเตอร์ของโจทก์และจำเลยว่ามีความเหมือนคล้ายในสาระสำคัญหรือไม่โดยใช้วิธีการ Abstraction Test ซึ่งเป็นวิธีการที่ศาลในคดี Nichols V. Universal Picture<sup>20</sup> เคยใช้และเป็นวิธีการที่ได้รับการยอมรับจากนักกฎหมายทั่วไป การทดสอบนี้เป็นการพิจารณาที่แยกความคิด(idea)ออกจากส่วนที่เป็นการแสดงออกของความคิด(expression of idea) เนื่องจากกฎหมายลิขสิทธิ์ไม่ให้การคุ้มครองในสิ่งที่เป็นเพียงแต่ความคิด หลังจากที่ได้พิจารณาแล้วผู้พิพากษาPratt ได้ตัดสินว่าโปรแกรม OSCAR 3.5 นั้นไม่มีความเหมือนคล้ายกับโปรแกรม ADAPTER ของโจทก์ นอกจากนี้ผู้พิพากษา Part ได้กล่าวถึงการพิจารณาของศาลในคดี Whelan ว่าเป็นการใช้เหตุผลที่ไม่ถูกต้อง เพราะการที่ศาลในคดีนี้ได้ขยายการคุ้มครองลิขสิทธิ์ไปถึงโครงสร้าง,การจัดลำดับขั้นตอนและองค์ประกอบของโปรแกรมคอมพิวเตอร์เป็นการบิดเบือนหลักของกฎหมายลิขสิทธิ์และแสดงให้เห็นว่าศาลยังไม่เข้าใจถึงลักษณะของงานโปรแกรมคอมพิวเตอร์ที่มีความสลับซับซ้อน โดยศาลเข้าใจผิดว่าความคิดในการเขียนโปรแกรมมีเพียงความคิดและเมื่อแยกความคิดออกไปแล้วส่วนที่เหลือ เช่น โครงสร้าง,การจัดลำดับขั้นตอน หรือ องค์ประกอบของโปรแกรม เป็นต้น เป็นการแสดงออกของความคิดที่กฎหมายลิขสิทธิ์ให้การคุ้มครอง

โจทก์ได้อุทธรณ์คำพิพากษาของศาลชั้นต้นต่อศาลอุทธรณ์ว่าโปรแกรม OSCAR 3.5 ละเมิดลิขสิทธิ์ในโปรแกรมคอมพิวเตอร์ของโจทก์และศาลชั้นต้นนั้นไม่เข้าใจในคำพิพากษาของศาลในคดี Whelan กับ Jaslow ที่ศาลในคดีนี้ให้การคุ้มครองไปถึงโครงสร้าง,การจัดลำดับขั้นตอนและองค์ประกอบของโปรแกรมคอมพิวเตอร์ จากการอุทธรณ์นี้ศาลอุทธรณ์ได้พิพากษายืนตามคำตัดสินของศาลชั้นต้นว่าถูกต้องและศาลได้ปฏิเสธที่จะใช้คำตัดสินตามแนวทางของศาลในคดี Whelan

แนวทางที่ศาลในคดี Computer Associates International, Inc. V. Altai Inc., ได้ใช้ในการพิจารณาถึงการละเมิดลิขสิทธิ์ในโปรแกรมคอมพิวเตอร์นั้นมี 3 ขั้นตอน คือ

1. Abstraction Test
2. Filtration Test
3. Comparison Test

ซึ่งผู้เขียนจะกล่าวถึงแนวทางในการพิสูจน์ 3 ขั้นตอนนี้อย่างละเอียดในบทที่ 5 ที่ว่าด้วยแนวทางในการพิสูจน์ของศาลต่างประเทศ

<sup>20</sup> 45 F.2d 119,121 (2d Cir. 1930) and 282 U.S. 902 (1931)

## ข. ประเทศญี่ปุ่น

ในประเทศญี่ปุ่นได้มีคดีที่เกี่ยวข้องกับประเด็นนี้คือ คดี System Science K.K. V. Toyo Sokuki K.K.<sup>21</sup> คดีนี้จะคล้ายกันคดี Computer Associates ของสหรัฐอเมริกาที่ได้ตัดสินถึงขอบเขตของการคุ้มครองสำหรับโปรแกรมคอมพิวเตอร์

ข้อเท็จจริงในคดีนี้มีว่าโจทก์ System Science ได้พัฒนาโปรแกรม 4 โปรแกรมคือ โปรแกรม MIC, ZA-FAII, Za-FMII และ CA-7II ที่เป็นโปรแกรมที่ใช้สำหรับการวัด Bio-chemical และจำเลยได้ก็อปโปรแกรมเหล่านี้โดยตรงยกเว้นก็แต่โปรแกรม CA-7II นอกจากนี้จำเลยได้ติดตั้งโปรแกรมเหล่านี้ลงบน ROM จึงศาลพบว่าการคัดลอกงานขึ้นมาใหม่นี้เป็นการละเมิดลิขสิทธิ์ของโจทก์

ประเด็นสำคัญในการพิจารณาจึงอยู่ที่โปรแกรม CA-7II ที่โจทก์กล่าวหาว่าจำเลยได้ดัดแปลงเพื่อสร้างโปรแกรมที่มีชื่อว่า CA-9 ศาลเห็นว่าโปรแกรม CA-7II ไม่ได้มีการแสดงออกซึ่งความคิดจึงเป็นสิ่งที่ไม่ได้รับการคุ้มครองตามกฎหมายลิขสิทธิ์ โดยศาลได้ให้เหตุผลว่า

1. โปรแกรม CA-9 ประกอบด้วยส่วนต่างๆ ที่เป็นพื้นฐานของโปรแกรม CA-7II
2. โปรแกรมย่อย(subroutine) ของ CA-7II เรียกคำสั่งที่สามารถถูกแสดงออกได้โดยมีข้อจำกัดในเรื่องของวิธีการ

ในคดีนี้โจทก์ได้ยื่นอุทธรณ์ โดยโจทก์ได้ยกประเด็นขึ้นมา 3 ประเด็น คือ

1. ถึงแม้ว่าโปรแกรม CA-9 ได้ใช้เพียงส่วนที่เป็นพื้นฐานของโปรแกรม CA-7II แต่ส่วนต่างๆ เหล่านี้ได้รับการคุ้มครองในฐานะที่ได้เป็นการแสดงออกซึ่งความคิดแล้ว
2. โปรแกรมย่อยได้มีการสร้างสรรค์ที่มีประสิทธิภาพเพียงพอที่เป็นผลมาจากความสามารถของนักเขียนโปรแกรมแล้ว จึงสมควรที่กฎหมายจะให้การคุ้มครอง
3. สามารถเห็นได้อย่างชัดเจนว่าโปรแกรมย่อยของ CA-7II และของ CA-9 มีอัลกอริทึมที่เหมือนกัน

<sup>21</sup> Tokyo Hight Court, 1322 Hanji 138 (June 20, 1989), code in, Jonathan Band and Masanobu Katoh, Interface on trial Intellectual property and interoperability in the global software industry, (San Francisco: Westview Press Inc., 1995) pp. 290-291.

ในคดีนี้ศาลสูงได้ปฏิเสธข้อโต้แย้งของโจทก์โดยได้มีการวิเคราะห์ดังนี้ “ In order to be able to decide whether a certain program infringes the copyright in a program work, it is of course necessary that there be a combination of instructions in the program work that can be found to be creative, and that the combination of instructions of the later-created program be similar to part of the program which can be found to be creative...”<sup>22</sup> เพราะฉะนั้นงานที่สร้างขึ้นมาภายหลังถึงจะเหมือนคล้ายกันบางส่วนก็สามารถเป็นงานที่มีลิขสิทธิ์ได้ถ้าเป็นการสร้างสรรค์งานด้วยตนเอง

ศาลได้พิจารณาถึงความเหมือนคล้ายของโปรแกรมทั้งสองตัวนี้ว่าเหมือนคล้ายกันอย่างไร โดยแยกพิจารณา ดังนี้

1. หน้าที่ของ การเปลี่ยนรูปแบบการวัด(measurement mode change), การใส่ข้อมูลโดยคีย์บอร์ด, การจัดแบ่งของพื้นที่การวัด และการเขียนลงในหน่วยความจำที่ทั้งหมดนี้จะถูกทำให้สำเร็จโดยฮาร์ดแวร์
2. การรวมกันของคำสั่งสำหรับโปรแกรมย่อยที่บังคับโดยฮาร์ดแวร์
3. โปรแกรมทั้งสองนี้พัฒนามาจากการรวมพื้นฐานของคำสั่งสำหรับโปรแกรมปฏิบัติการย่อยเมื่อการปฏิบัติการของเครื่องพิมพ์ไม่สามารถปฏิบัติงานได้
4. หลักทั่วไปซึ่งถึงการตั้งโปรแกรมย่อยโดยเรียงซ้อนกันที่ส่วนเก็บระบบปฏิบัติการที่เป็นส่วนหนึ่งของฮาร์ดแวร์

จากข้อพิจารณาดังกล่าวศาลได้ข้อสรุปว่าไม่พบการละเมิดลิขสิทธิ์แม้ว่าจะมีความเหมือนคล้ายกันก็ตาม เพราะส่วนที่เหมือนคล้ายกันนั้นมิใช่ในการแสดงออกที่ถูกจำกัดโดยฮาร์ดแวร์, การแสดงออกนั้นเป็นการแสดงออกที่เป็นพื้นฐาน และ ยังเป็นการแสดงออกที่ถูกสั่งโดยความคิดทั่วไป(Common sense) เนื่องจากว่าสิ่งเหล่านี้ไม่ได้รับการคุ้มครองตามกฎหมายลิขสิทธิ์ นอกจากนั้นศาลยังยืนยันว่าผังการปฏิบัติงาน (processing flow) ไม่ได้รับการคุ้มครองเนื่องจากมีลักษณะเป็นอัลกอริทึมที่เป็นสิ่งที่กฎหมายลิขสิทธิ์ของญี่ปุ่นกำหนดไว้อย่างชัดเจนในมาตรา 10 (3)ว่าไม่ให้การคุ้มครองไปถึงสิ่งที่เป็นอัลกอริทึม

นอกจากนี้ยังมีคดี ICM Corp. V. Met's, Inc.<sup>23</sup> ในคดีนี้ ICM กล่าวหาว่า File HCA ของ Met's ละเมิดลิขสิทธิ์ใน Install Batch Files (IBF file) ของโปรแกรม EO System ของ

<sup>22</sup> Ibid., p. 140.

<sup>23</sup> Tokyo High Court 24-1, Chisai-shu 218 (Mar. 31, 1992), code in, Jonathan Band and Masanobu Katoh, Interface on trial Intellectual property and

โจทก์ ซึ่ง IBF file อธิบายถึงคำสั่งและข้อมูลที่จำเป็นสำหรับการติดตั้งโปรแกรมประยุกต์ลงบนฮาร์ดดิสก์

คดีนี้โจทก์อ้างว่าแฟ้ม IBF เป็นโปรแกรมที่ได้รับการคุ้มครองโดยกฎหมายลิขสิทธิ์ เพราะเป็นผลจากการสร้างสรรค์ และแฟ้ม HCA เป็นการก๊อปปี้แฟ้ม IBF จำเลยโต้แย้งว่าแฟ้ม IBF เป็นเพียงแฟ้มข้อมูลที่ไม่ได้รับการคุ้มครองลิขสิทธิ์ในฐานะที่เป็นงานโปรแกรม (program work) และแฟ้ม IBF ไม่ได้เป็นผลมาจากการสร้างสรรค์ที่จำเป็นที่จะต้องได้รับการคุ้มครองโดยกฎหมายลิขสิทธิ์ ศาลได้ตัดสินว่าแฟ้ม IBF ไม่เป็นงานที่มีความสร้างสรรค์พอที่จะได้รับการคุ้มครอง และปฏิเสธที่จะสรุปว่าแฟ้ม IBF เป็นงานโปรแกรมภายใต้กฎหมายลิขสิทธิ์ รวมทั้งศาลยังตัดสินว่าตัวอักษรและสัญลักษณ์ที่ใช้ในแฟ้ม IBF ไม่ได้รับการคุ้มครอง เพราะว่าเป็นภาษาที่ใช้เขียนโปรแกรม (programming language) ภายใต้มาตรา 10(3)(1) นอกจากนี้ศาลยังพบว่าการแสดงออกในแฟ้ม IBF ถูกจำกัดโดยแฟ้ม MENU.EXE และไม่มีทางเลือกสำหรับการเลือกของการแสดงออก ไม่สามารถมีการสร้างสรรค์ (creativity) ในแฟ้ม IBF ที่ศาลไม่ให้การคุ้มครองเพราะว่าการแสดงออกนั้นถูกจำกัดโดยปัจจัยภายนอก เช่น แฟ้มต่างๆ และ โปรแกรมที่แฟ้ม IBF ติดต่อกับงานอยู่ (interoperation)

ในคดีนี้ศาลได้ตัดสินคล้ายกับคดี Computer Associate ที่ศาลได้แบ่งแยกโปรแกรมคอมพิวเตอร์ของโจทก์ออกเป็นส่วนต่างๆ กันว่าส่วนใดเป็นความคิดและการแสดงออกซึ่งความคิด ซึ่งเป็นขั้นตอนที่ในคดี Computer Associate เรียกว่า abstraction และได้วิเคราะห์ถึงส่วนใดบ้างที่สามารถได้รับการคุ้มครอง ซึ่งเทียบได้กับขั้นตอน filtration ของคดี Computer Associate และในคดีนี้ศาลได้ตัดสินว่าส่วนที่ได้รับการคุ้มครองโดยข้อจำกัดภายนอก ได้แก่ ความต้องการต่อการติดต่อกับโปรแกรมอื่น เป็นต้น

จึงเห็นได้ว่าโปรแกรมคอมพิวเตอร์ในส่วนที่เป็นหลักการพื้นฐานและเป็นสิ่งที่นักเขียนโปรแกรมคอมพิวเตอร์ทั่วไปใช้ในการเขียนโปรแกรมในฐานะที่เป็นหลักการพื้นฐานในการเขียนโปรแกรมจะเป็นสิ่งที่กฎหมายลิขสิทธิ์ของประเทศญี่ปุ่นไม่ให้การคุ้มครอง ดังนั้นในโครงสร้าง (structure), การจัดลำดับขั้นตอน (sequence) และการจัดวางองค์ประกอบ (organization) ของโปรแกรมคอมพิวเตอร์ซึ่งเป็นหลักการพื้นฐานในการเขียนโปรแกรมจึงไม่ได้รับการคุ้มครองตามกฎหมายลิขสิทธิ์ของญี่ปุ่น สิ่งเหล่านี้ถือว่าเป็นหลักการพื้นฐานเพราะว่าในการเขียน

โปรแกรมจะมีข้อจำกัดทางเทคนิคหลายประการที่ทำให้วิธีการในการเขียนโปรแกรม หรือ โครงสร้าง หรือส่วนประกอบต่างๆ ของโปรแกรมจำกัด จึงเป็นผลให้โปรแกรมสองโปรแกรมอาจเหมือนกันได้โดยไม่มี การลอกเลียนกัน

### ค. กลุ่มประชาคมยุโรป

ในเรื่องของโครงสร้าง (structure), การจัดลำดับขั้นตอน (sequence) และการจัดวางองค์ประกอบ (organization) ของโปรแกรมคอมพิวเตอร์ซึ่งเป็นหลักการพื้นฐานในการเขียนโปรแกรมถือได้ว่าเป็นหลักการ ซึ่งในข้อบังคับของ EC นั้นอัลกอริทึม (algorithms), ตรรกะ (logic) หรือภาษาเขียนโปรแกรม (programming languages) เป็นการผสมกันระหว่างความคิดและหลักการ โดยที่ความคิดและหลักการ (ideas and principles) ไม่สามารถได้รับการคุ้มครองตามที่กำหนดไว้ในข้อบังคับมาตราที่ 1(2) ดังนั้นจึงไม่ได้รับการคุ้มครองตามข้อบังคับนี้ ในตอนที่ร่างข้อบังคับนี้ได้มีกลุ่มของบริษัทคอมพิวเตอร์ที่ใหญ่ๆ ของสหรัฐอเมริกา เช่น IBM, DEC และ Apple ได้พยายามที่จะสนับสนุนให้ข้อบังคับให้การคุ้มครองแก่ structure, sequence และ organization ของโปรแกรมคอมพิวเตอร์เพื่อประโยชน์แก่บริษัทของตนที่จะมีสิทธิผูกขาดในงานนั้นแต่เพียงผู้เดียว<sup>24</sup> แต่การผลักดันนี้ก็ยุติลงที่ว่า ข้อบังคับนี้ไม่ให้การคุ้มครองไปถึงโครงสร้าง (structure), การจัดลำดับขั้นตอน (sequence) และการจัดวางองค์ประกอบ (organization) ของโปรแกรมคอมพิวเตอร์

### ง. ประเทศไทย

ในพระราชบัญญัติลิขสิทธิ์บัญญัติในมาตรา 6 วรรคสองแต่เพียงว่า การคุ้มครองลิขสิทธิ์ไม่คลุมถึงความคิดหรือขั้นตอน กรรมวิธีหรือระบบ หรือวิธีการใช้งาน หรือแนวความคิด หลักการ การค้นพบ หรือทฤษฎีทางวิทยาศาสตร์ หรือคณิตศาสตร์ โดยที่บทบัญญัตินี้ครอบคลุมงานอันมีลิขสิทธิ์ในทุกประเภทซึ่งรวมถึงโปรแกรมคอมพิวเตอร์ด้วย เนื่องจากในดวบทไม่การอธิบายมากไปกว่านี้และในประเทศไทยก็ยังไม่มีการพิพากษาของศาลที่ตัดสินในเรื่องนี้ แต่ผู้เขียนเห็นว่าถ้ามีคดีเกิดขึ้นศาลไทยก็คงตัดสินตามแนวทางที่เป็นอยู่ของสหรัฐอเมริกาหรือของกลุ่มประชาคมยุโรป ที่ว่าโครงสร้าง (structure), การจัดลำดับขั้นตอน (sequence) และการจัดวางองค์ประกอบ (organization) ของโปรแกรมคอมพิวเตอร์นั้นเป็นหลักการพื้นฐานในการเขียน

<sup>24</sup> Pamela Samuelson, "Comparing U.S. and EC copyright protection for computer programs: Are they more different than they seem?," *Journal of law and commerce*, 13 (1994): 290-291.

โปรแกรมจึงไม่ได้รับการคุ้มครองตามกฎหมายลิขสิทธิ์ เพราะในมาตรา 6 วรรคสองได้เปิดกว้างให้ศาลใช้ดุลยพินิจในการตัดสินว่าอะไรที่เข้าข่ายที่กฎหมายลิขสิทธิ์จะไม่ให้การคุ้มครอง

### 3.3 การกระทำที่เป็นการดัดแปลงในตัวเชื่อมประสานระหว่างคอมพิวเตอร์กับผู้ใช้ ( user Interface )

#### 3.3.1 ความหมายของตัวเชื่อมประสานระหว่างคอมพิวเตอร์กับผู้ใช้ ( user interface )

การที่ต้องการคำนึงถึงตัวเชื่อมประสานระหว่างเครื่องคอมพิวเตอร์กับผู้ใช้เพราะเนื่องมาจากในการแข่งขันทางการค้าบุคคลที่จะประสบความสำเร็จสินค้าของเขาจะต้องได้รับความพึงพอใจจากลูกค้าและสินค้านั้นสามารถตอบสนองความต้องการของผู้บริโภคได้มากที่สุด สินค้าชิ้นนั้นจึงจะประสบความสำเร็จในการตลาด เครื่องคอมพิวเตอร์และโปรแกรมคอมพิวเตอร์ก็เช่นเดียวกัน ส่วนมากความต้องการของผู้ใช้ที่รวบรวมได้มีดังต่อไปนี้คือ<sup>25</sup>

1. ระยะเวลาในการเรียนรู้ เนื่องจากถ้าชุดคำสั่งใดที่ผู้ใช้สามารถเรียนรู้การใช้งานได้ในระยะเวลาที่สั้นผู้ใช้ก็จะเกิดความรู้สึกที่มีความสะดวกสบายในการใช้มากกว่าการใช้ชุดคำสั่งที่ต้องเสียเวลาในการเรียนรู้ที่นานมากกว่า

2. ความเร็วในการตอบสนอง ในการใช้งานเครื่องคอมพิวเตอร์นั้นระหว่างที่เราใช้จะต้องคอยการตอบสนองหรือคำตอบจากเครื่องคอมพิวเตอร์ซึ่งถ้ายังไม่เห็นผลลัพธ์ออกมา ก็ยังไม่สามารถรู้ได้ว่ามีการผิดพลาดในการทำงานหรือไม่ยอมทำให้ผู้ใช้เกิดความไม่พึงพอใจได้ ดังนั้นความรวดเร็วในการตอบสนองจึงเป็นสิ่งสำคัญประการหนึ่งในการสร้างความพึงพอใจให้กับผู้ใช้

3. อัตราความผิดพลาดที่เกิดในระหว่างการใช้งาน ในระหว่างการใช้งานถ้าเกิดมีข้อผิดพลาดเกิดขึ้นอยู่เสมอ ซึ่งข้อผิดพลาดส่วนมากจะเกิดจากชุดคำสั่งที่ผู้ใช้ได้เลือกโดยผู้ใช้อาจใส่ข้อมูลผิดหรือทำงานผิดขั้นตอน เมื่อเกิดข้อผิดพลาดขึ้นแม้ว่าจะเกิดจากความผิดพลาดของตัวเองผู้ใช้อาจเกิดความไม่พึงพอใจได้ ดังนั้นโปรแกรมที่ดีนั้นต้องมีการป้องกันการผิดพลาดโดยการมีข้อแนะนำที่ชัดเจนในแต่ละขั้นตอนการใช้งานและการออกแบบควรให้ผู้ใช้เข้าใจการทำงานได้อย่างสะดวก

4. ระดับความพึงพอใจของผู้ใช้ ส่วนมากจะวัดโดยการประเมินเป็นระดับ เช่นระดับ A, B, C เป็นต้น ดังนั้นในการเขียนโปรแกรมผู้เขียนจะต้องคำนึงถึงความพึงพอใจและความต้องการของผู้ใช้เป็นสำคัญ

<sup>25</sup> Shneiderman Ben, Design the user interface: Strategies for effective human-computer interaction(Addision-Wesley Publishing, 1986), pp. 14-16.

5. ความสะดวกและการจดจำที่ง่ายในการใช้งาน เนื่องจากในการใช้งานถ้าโปรแกรมนั้นใช้งานได้ง่ายผู้ใช้อย่อมเลือกที่จะใช้โปรแกรมนั้นมากกว่าโปรแกรมที่มีความยากมากในการใช้งาน

### 3.3.2 ขอบเขตการคุ้มครองตัวเชื่อมประสานระหว่างโปรแกรมคอมพิวเตอร์กับผู้ใช้ในประเทศต่าง ๆ

#### ก. ประเทศสหรัฐอเมริกา

คอมพิวเตอร์ซอฟต์แวร์ในปัจจุบันนี้พยายามรักษาความสมดุลระหว่างอัลกอริทึมที่เป็นหลักของโปรแกรมและประสบการณ์ของผู้ใช้เมื่อได้มีการใช้งานโปรแกรม ตลาดซอฟต์แวร์ในปัจจุบันนี้มีการแข่งขันที่สูงมากโปรแกรมที่จะสามารถดำรงอยู่ได้ในตลาดต้องไม่เพียงแต่มีลักษณะของโปรแกรมคอมพิวเตอร์ที่มีระบบอย่างสมบูรณ์แล้ว แต่โปรแกรมคอมพิวเตอร์นั้นต้องให้ความรู้สึกที่สะดวกสบายแก่ผู้ใช้หรือที่เรียกกันว่า user friendly ดังนั้นเรื่อง look and feel ของโปรแกรมจึงเป็นจุดที่สำคัญในความสำเร็จของโปรแกรม การที่สิ่งนี้เป็นประเด็นสำคัญสำหรับการพัฒนาโปรแกรมคอมพิวเตอร์เพราะเมื่อโปรแกรมคอมพิวเตอร์แบบใดที่ให้ความสะดวกสบายแก่ผู้ใช้และเป็นที่ยอมรับของผู้ใช้ก็ย่อมเป็นโปรแกรมที่ครองตลาดได้มากกว่าโปรแกรมอื่น ดังนั้นจึงเห็นได้ว่าโปรแกรมที่จะประสบความสำเร็จนั้นต้องมีลักษณะของการใช้งานที่ผู้ใช้ได้รับความสะดวกสบายไม่ว่าจะเป็นการจัดลำดับ, ขั้นตอนในการทำงาน, เมนูคำสั่งต่าง ๆ, การปรากฏทางหน้าจอ, ภาพกราฟฟิกต่าง ๆ และการแสดงข้อมูลต่าง ๆ เป็นต้น สิ่งเหล่านี้อาจเรียกโดยรวมว่า Graphical User Interface (GUIs)

ดังนั้นจึงเป็นเหตุให้บริษัทผู้ผลิตซอฟต์แวร์อื่นพยายามที่จะพัฒนาโปรแกรมของตนให้ผู้ใช้มีความรู้สึกที่สะดวกสบายเวลาใช้ ซึ่งเมื่อพัฒนาเสร็จแล้วปรากฏว่าลักษณะของโปรแกรมคอมพิวเตอร์และการติดต่อระหว่างผู้ใช้กับเครื่องคอมพิวเตอร์มีลักษณะและให้ความรู้สึกที่เหมือนกันคล้ายกับโปรแกรมแรกที่ได้มีการออกวางสู่ตลาด จึงทำให้บริษัทแรกที่ผลิตโปรแกรมออกมาสูญเสียผลประโยชน์ในตลาดไปและจึงได้ฟ้องบริษัทที่ได้พัฒนาโปรแกรมคอมพิวเตอร์ใหม่มาว่าละเมิดลิขสิทธิ์ในโปรแกรมคอมพิวเตอร์ของตน ดังเช่นคดีที่จะได้กล่าวต่อไปนี้

ในสหรัฐอเมริกามีแนวคำพิพากษาที่เกี่ยวข้องเรื่อง Look and Feel อยู่ 2 แนวทาง คือ



1. แนวที่ให้การคุ้มครองการออกแบบหน้าจอ หรือ ตัวเชื่อมประสาน(user interface) ที่เป็นการขยายการคุ้มครองของกฎหมายลิขสิทธิ์ ซึ่งได้มีคดีที่ศาลตัดสินในแนวนี้คือ คดี Broderbund V. Unison World, คดี Degital Communications V. Softklone , คดี Lotus V. Paperback, คดี Lotus V. Mosaic

2. แนวทางที่ผู้พิพากษาได้จำกัดขอบเขตของการคุ้มครองการออกแบบหน้าจอ หรือ ตัวเชื่อมประสาน (user interface) เพื่อไม่ให้ส่งผลกระทบต่อทางธุรกิจและประโยชน์ของสังคม ส่วนรวม ซึ่งได้มีคดีที่ศาลได้ตัดสินในแนวนี้คือ คดี Synercome Tecnology V. University Computing, คดี Plain Cotton V. Goodpasture Computer, คดี Apple V. Microsoft and Hewlett-Packard, คดี Lotus V. Borland

คดีที่ศาลได้มีการขยายขอบเขตการคุ้มครองให้คุ้มครองถึง user interface นั้น ได้มีรายละเอียดและข้อเท็จจริงดังต่อไปนี้

คดี Broderbund Software, Inc., V. Unison World, inc.,<sup>26</sup> คดีนี้เป็นคดีเริ่มแรกของคดีที่เกี่ยวกับ look and feel ข้อเท็จจริงในคดีนี้มีว่า David Balsam และ Martin Kahn ได้นำ Quicky program ที่เขาได้พัฒนาให้ Broderbund Software, Inc., ซึ่งเป็นบริษัทซอฟต์แวร์ที่แคลิฟอร์เนียที่เชี่ยวชาญในโปรแกรมคอมพิวเตอร์ส่วนบุคคล โดยความหวังที่ว่า Broderbund Software, Inc., จะนำไปจำหน่ายเพื่อตน โปรแกรมนี้จะสร้างการด้อยพรบนแผ่นดินสิทธิ์แก่ความคิดที่ว่าประชาชนจะส่งสิทธิ์แก่บุคคลอื่นโดยทางไปรษณีย์ การแลกเปลี่ยนกันในการอวยพรวันเกิดและเทศกาลต่างๆต่อบุคคลอื่นโดยผ่านหน้าจอคอมพิวเตอร์ของเขา Borderbund ได้แนะนำให้ Balsam และ Kahn เขียนโปรแกรมของเขาขึ้นมาใหม่อีกครั้งหนึ่ง เพราะว่าโปรแกรมจะเตรียมการด้อยพรที่พิมพ์ออกมาและเอกสารกราฟฟิกอื่นๆ เช่น เครื่องหมายต่างๆ เป็นต้น Borderband ได้ทำงานร่วมกับ Balsam และ Kahn เกือบทั้งปีเพื่อพัฒนาโปรแกรมซึ่งจะเริ่มออกสู่ตลาดในชื่อ "Print Shop" ที่ใช้สำหรับเครื่องคอมพิวเตอร์ Apple ในเดือนพฤษภาคม ปี ค.ศ. 1984

ในเดือนเดียวกันนั้นประธานของ Borderbund ได้พบกันประธานของบริษัท Unison World คือ Mr. Hong Lu ที่ทำธุรกิจการเปลี่ยนโปรแกรมสำหรับที่ใช้เครื่องหนึ่งให้ไปใช้งานบนอีกเครื่องหนึ่งได้ซึ่งส่วนใหญ่เขาจะดัดแปลงโปรแกรมคอมพิวเตอร์มากกว่าที่จะเขียนขึ้นมาใหม่ทั้งหมด จากการประชุม Unison สนใจที่จะสร้างโปรแกรมของ Print Shop ให้ใช้สำหรับเครื่อง

<sup>26</sup> 648 F. Supp. 1127 (N.D. Cal. 1986), อ้างจาก Anthony Lawrence Clapes, *Software, copyright, and competition: The "look and feel" of the law*, pp.145-151.

IBM PC บุคคลากรของ Unison ได้ไปพบกับ Balsam และ Kahn เพื่อตรวจสอบข้อเท็จจริงที่จะผลิตงานขึ้นมาใหม่ Balsam ได้แสดงโปรแกรมต้นฉบับ(source code)ของโปรแกรม Print Shop ต่อนักเขียนโปรแกรมของ Unison และได้ให้แผ่นดิสก์ที่ได้ก็อปปีโปรแกรมนี้ไว้หลายอันต่อ MacDuff Hughes เพื่อที่ทาง Unison จะได้แสดงต่อผู้ผลิตซอฟต์แวร์ชาวญี่ปุ่นที่ทาง Unison จะทำสัญญาให้เขาทำการพัฒนา

Hughes ได้พบกับ Balsam และ Kahn หลายครั้งเพื่อถกปัญหาถึงความก้าวหน้าของเขา แต่การตกลงสำหรับการอนุญาตนี้ได้จบลงในขณะที่ Hughes ได้ทำงานถึงครึ่งหนึ่งแล้ว คู่สัญญาได้ยุติกันแต่ Hughes ไม่หยุดการทำงาน และเขาได้ผลิตโปรแกรมจนสำเร็จและออกสู่ตลาดในชื่อว่า Printmaster ในเดือนมีนาคม ค.ศ. 1985 และต่อมาอีก 2 เดือนบริษัท Broderbund ได้ฟ้องร้องบริษัท Unison

ในศาลชั้นต้นนี้ผู้พิพากษาที่พิจารณาคือ ผู้พิพากษา William Orrick ปัญหาแรกที่ศาลต้องพิจารณาคือ การติดต่อทางหน้าจอ (screen interface)ของคอมพิวเตอร์ได้รับการคุ้มครองโดยกฎหมายลิขสิทธิ์หรือไม่ ศาลได้ให้คำตอบว่าได้รับการคุ้มครอง Unison ได้โต้แย้งว่า หน้าจอของ Print Shop ที่แสดงออกมาไม่มีอะไรแตกต่างไปจากความคิดที่ได้วางไว้ที่ menu-driver , โปรแกรมสำหรับการ์ดที่พิมพ์, สัญลักษณ์ และความเหมือนใน การติดต่อทางหน้าจอที่เหมือนกันในสาระสำคัญกับโปรแกรม Print Shop เพราะว่าไม่มีหนทางอื่นที่เป็นไปได้ของโครงสร้างโปรแกรมดังนั้นจึงจำเป็นที่จะต้องเหมือนกัน แต่ผู้พิพากษาได้เห็นด้วยกับคดี Whelan โดยให้การคุ้มครองแก่การติดต่อทางหน้าจอและปฏิเสธข้อต่อสู้ของ Unison ไว้ 2 ประการที่ว่า หน้าจอ(screen)เป็นงานอรรถประโยชน์และกฎหมายลิขสิทธิ์ไม่ให้การคุ้มครองงานทางรูปภาพหรือกราฟฟิกที่เป็นอรรถประโยชน์ และข้อต่อสู้ที่ว่าหน้าจอเป็นสิ่งพื้นๆ ของกฎและคำสั่งที่กฎหมายลิขสิทธิ์ไม่ให้การคุ้มครองถึงกฎและคำสั่ง (rule and instruction)

ในคดีนี้ศาลได้พิพากษาว่า Unison ได้ละเมิดลิขสิทธิ์ในโปรแกรมของ Borderbund โดยการลอกเลียนโครงสร้าง, การจัดลำดับ และแผนผังของ user interface ของโปรแกรม Print Shop นอกจากนั้นศาลไม่ต้องการที่จะแยกลิขสิทธิ์ทางโสตทัศนวัสดุ (audiovisual) ที่ได้รับสำหรับหน้าจอ(screen)

คดีต่อไปที่ให้การคุ้มครองแก่การติดต่อทางหน้าจอ (screen interface) คือคดี Digital Communications Associates, Inc., V. Softklone Distributing Corp.,<sup>27</sup>

<sup>27</sup> 659 F. Supp. 449, 2 U.S.P.Q. 22 (1987) N.D. a.1987

คดีนี้ศาลได้เสื่อมคลายลงจากกฎของผู้พิพากษา Qrrick ในการพิจารณา และยึดถือว่า ลิขสิทธิ์ในสื่อทัศนวัสดุ(audiovisual)ที่แยกออกมา ต้องถูกจดทะเบียนสำหรับหน้าจอ(screen) เพื่อการฟ้องร้องต่อการละเมิดลิขสิทธิ์ของสิ่งเหล่านี้ ไม่เพียงแต่หน้าจอที่ได้รับการคุ้มครองโดย กฎหมายลิขสิทธิ์ในโปรแกรมตามที่ผู้พิพากษา Qrrick ได้ตัดสินเท่านั้น แต่สำนักงานลิขสิทธิ์จะไม่ยอมรับสิ่งที่แยกสำหรับการแสดงทางหน้าจอ(display screen) เพื่อการฟ้องร้องสำหรับการ ละเมิดในการแสดงทางหน้าจอ เขาต้องพิสูจน์ว่าได้มีการจดทะเบียนในหน้าจอไม่เพียงเฉพาะ การจดทะเบียนในโปรแกรม อย่างไรก็ตามสำนักงานลิขสิทธิ์จะไม่จดทะเบียนลิขสิทธิ์ในหน้าจอ แต่เพียงอย่างเดียวจะต้องจดทะเบียนโปรแกรมคอมพิวเตอร์ด้วยเพราะเชื่อว่าหน้าจอได้รับการ คุ้มครองโดยกฎหมายลิขสิทธิ์ในส่วนของโปรแกรมแล้ว<sup>28</sup>

คดีต่อไปคือคดี Lotus Development Corp., V. Paperback Software International<sup>29</sup> ในคดีนี้จำเลยคือบริษัท Paperback Software International ได้พัฒนาโปรแกรม Spreadsheet ที่มีชื่อว่า V.P.Planner โดยที่ผู้เขียนโปรแกรมนี้ได้ใช้แนวทางในการเขียนโปรแกรมให้เมนู สามารถทำงานร่วมกับโปรแกรม Lotus1-2-3 ของโจทก์ได้ แม้ว่าโปรแกรมของจำเลยจะใช้ คำสั่ง, ภาษา Macro (Macro language), วากยสัมพันธ์(syntax) \*, โครงสร้างและการจัดลำดับ ของเมนูและข้อความเหมือนกับโปรแกรม Lotus ของโจทก์ แต่โปรแกรมของจำเลยได้ดัดแปลง รายละเอียดของเมนูคำสั่งต่าง ๆ และได้เปลี่ยนจุดวางคำสั่งจากด้านบนสุดของหน้าจอ คอมพิวเตอร์ที่เป็นแบบที่ Lotus1-2-3 ได้ใช้อยู่มาไว้ที่ส่วนล่างสุดของหน้าจอ ในคดีนี้ผู้ พิพากษา Robert Keeton ได้พิพากษาว่า เมื่อได้มีการพิจารณาถึงลักษณะโดยรวมของหน้าจอ ของโปรแกรมของโจทก์และจำเลยมีความเหมือนคล้ายกันอย่างมากถึงแม้ว่าจะมีรายละเอียด ปลีกย่อยบางประการที่แตกต่างกันบ้าง ศาลก็ถือว่าจำเลยได้ละเมิดลิขสิทธิ์ในการออกแบบหน้า จอของโจทก์

<sup>28</sup> Anthony Lawrence Clapes, Software, copyright, and competition: The "look and feel" of the law, p. 151.

<sup>29</sup> 740 F. Supp. 37 (D. Mass. 1990)

\* วากยสัมพันธ์ (syntax) หมายถึง กฎเกณฑ์ในการผูกประโยค ในภาษาคอมพิวเตอร์ ก็ต้องมีวากยสัมพันธ์เหมือนภาษามนุษย์ธรรมดาเช่นกัน

ทักษิณา สวานานนท์, ศัพท์คอมพิวเตอร์สำหรับผู้ปฏิบัติงานแมคอินทอชและพีซี, หน้า 317.

ในส่วนต่อไปนี้จะกล่าวถึงคดีที่ศาลได้ตัดสินว่าไม่ให้ขายของเขตของการคุ้มครองไปถึง user interface คดีแรกคือ คดี Plains Cotton Cooperative Association V. Goodpasture Computer Service Inc.,<sup>30</sup>

ข้อเท็จจริงในคดีนี้มีว่า Plains Cotton เป็นสหกรณ์ของชาวไร่ฝ้ายในรัฐ Texas และรัฐ Oklahoma เพื่อช่วยเหลือสมาชิกในการปลูกและการตลาดฝ้าย Plains Cotton ได้พัฒนาโปรแกรมคอมพิวเตอร์ชื่อว่า Telcot ที่ใช้ช่วยเหลือสมาชิกในการให้ข้อมูลเกี่ยวกับราคาฝ้ายและความต้องการขาย, ให้บริการทางด้านบัญชีและช่วยเหลือในการทำสัญญาซื้อขายให้สำเร็จ Telcot ได้ออกสู่ตลาดในปี 1975 ซึ่ง 4 ปีต่อมาผู้จัดการสหกรณ์ได้ยุติโปรแกรมนี้ออกไปเพื่อเปิดโอกาสให้บริษัทอื่นเข้ามาปรับปรุง Telcot สำหรับเครื่องคอมพิวเตอร์ส่วนบุคคล บริษัท CXS ได้เข้ามาทำความเข้าใจกับสหกรณ์สำหรับการพัฒนาและการตลาดของรุ่น Telcot ที่สามารถใช้กับเครื่องคอมพิวเตอร์ส่วนบุคคลซึ่งจะเป็นของ CXS และสหกรณ์

วันที่ 13 เมษายน ค.ศ. 1985 CXS ได้ว่าจ้างนักเขียนโปรแกรมของสหกรณ์ 4 คนซึ่งเป็นผู้มีส่วนช่วยพัฒนา Telcot และหนึ่งในนักเขียนโปรแกรมนี้ออกจากสหกรณ์พร้อมกับได้นำโปรแกรมต้นฉบับ(source Code)ของ Telcot ไปด้วย และเมื่อวันที่ 18 เมษายน ค.ศ. 1985 สหกรณ์ได้ยุติสัญญากับบริษัท CXS

ลูกจ้าง 4 คนก็ยังคงทำงานกับบริษัท CXS ต่อและภายในปีนั้น CXS เนื่องจากค่าใช้จ่ายของบริษัท CXS มีมากจนกระทั่งถึงขั้นล้มละลายในเดือนมีนาคม ค.ศ. 1985 ในขณะที่บริษัท Goodpasture Computer Service ได้สนใจในโครงการ PC-Telcot และได้ว่าจ้างนักเขียนโปรแกรมทั้งหมดไว้ ถึงแม้ว่านักเขียนโปรแกรมทั้ง 4 คนจะได้ลงนามในสัญญาว่าจะไม่เปิดเผยความลับของนายจ้างคนก่อน นอกจากนั้นหนึ่งในนักเขียนโปรแกรม 4 คนได้แผ่นดิสก์ที่ได้บรรจุ Telcot ที่ตนได้พัฒนาอยู่นี้ต่อ บริษัท Goodpasture Computer Service

หลังจากที่ได้รับความร่วมมือของบริษัท Goodpasture Computer Service และนักเขียนโปรแกรม 4 คนนั้นก็ทำได้ทำโปรแกรมสำเร็จโดยใช้ชื่อว่า GEMS เมื่อวันที่ 15 มกราคม ค.ศ. 1986 สหกรณ์ได้ฟ้องบริษัท Goodpasture Computer Service และนักเขียนโปรแกรม 4 คนในข้อหาว่าได้ละเมิดลิขสิทธิ์ในโปรแกรมของตน ในคดีนี้ผู้พิพากษาศาลชั้นต้น ที่มีหน้าที่ตัดสินคดีคือ ผู้พิพากษา Halbert O. Woodward ได้สรุปว่า สหกรณ์ไม่มีเหตุผลที่จะประสบความสำเร็จ

<sup>30</sup> 807 F. 2d, 1256 (5th Cir. 1987) , code in, Anthony Lawrence Clapes, *Software, copyright, and competition: The "look and feel" of the law*, pp. 152-160.

สำเร็จในการกล่าวหาถึงการละเมิดลิขสิทธิ์ และปฏิเสธความคิดของสหกรณ์ ดังนั้นสหกรณ์จึงได้ยื่นอุทธรณ์ต่อศาลอุทธรณ์ที่ 5

ศาลอุทธรณ์ได้อธิบายถึงข้อเท็จจริงว่าโปรแกรม GEMS เหมือนกับโปรแกรม Telcot บนระดับของ Functional specification, การเขียนโปรแกรม(programming) และเอกสารประกอบการเขียนโปรแกรม (documentation) ความแตกต่างที่สำคัญระหว่างสองโปรแกรมอยู่ที่ว่า Telcot ถูกออกแบบเพื่อทำงานบนเครื่องคอมพิวเตอร์เมนเฟรม แต่ GEMS ถูกออกแบบสำหรับใช้งานในเครื่องคอมพิวเตอร์ส่วนบุคคล

ในชั้นศาลอุทธรณ์ สหกรณ์ได้โต้แย้งว่า ผู้พิพากษา Woodward ปรับใช้มาตรฐานที่ผิดสำหรับการตัดสินการละเมิดลิขสิทธิ์ สหกรณ์ยืนยันว่า ผู้พิพากษาของศาลชั้นต้นตัดสินเพียงว่าไม่มีหลักฐานของการคัดลอกบรรทัดต่อบรรทัดหรือการคัดลอกตัวอักษรและคำสแลงที่จะพิจารณาถึงหลักฐานของการคัดลอกการจัดองค์ประกอบ (organization copying) อย่างเช่นในคดี Whelan

ศาลอุทธรณ์ไม่เห็นด้วยกับข้อโต้แย้งของสหกรณ์ ผู้เชี่ยวชาญได้ทดสอบแล้วว่า Telcot ที่เป็นโปรแกรมสำหรับเครื่องคอมพิวเตอร์เมนเฟรมเป็นสิ่งที่ยากมากในการปรับปรุงให้นำมาใช้กับเครื่องคอมพิวเตอร์ส่วนบุคคลได้ การที่นักเขียนโปรแกรม 4 คนมาเขียนโปรแกรม GEMS ในโปรแกรม GEMS ได้ถูกเขียนโดยรากฐานบนความรู้และประสบการณ์ของนักเขียนโปรแกรมทั้ง 4 คนในอุตสาหกรรมฝ่าย ศาลได้ยึดหลักในคดี Synercom ที่ว่าการจัดลำดับ(sequencing) และการสั่ง (ordering) ของรูปแบบการใส่ข้อมูลของโปรแกรม STRAN (STRAN's input format) ไม่สามารถก่อให้เกิดการแสดงออกซึ่งความคิดที่กฎหมายลิขสิทธิ์ให้การคุ้มครองเพราะว่าการจัดลำดับ(sequencing)และการสั่ง(ordering)ไม่แสดงความคิดอื่นๆ แยกออกจากการจัดลำดับ (sequencing) และ การสั่ง(ordering)ในตัวของมันเอง ผู้พิพากษา Higginbotham ที่เป็นผู้ตัดสินคดี Synercom นี้มีความเห็นว่า การจัดรูปแบบการบันทึก (format) คอมพิวเตอร์นั้นมีทางเลือกที่เป็นไปได้หลายทางมากกว่าทางเลือกของการออกแบบเกียร์ที่วางตำแหน่ง 1,2,3,4 ที่เป็นรูปตัว H และถ้าให้การคุ้มครองการออกแบบเกียร์นี้ย่อมเท่ากับเป็นการให้สิทธิผูกขาดแก่ผู้ที่ประดิษฐ์คนแรกและทำให้ผู้แข่งขันไม่สามารถแข่งขันได้เพราะบุคคลทั่วไปกดดันในรูปแบบของเกียร์แบบนี้ พื้นฐานของการสรุปของผู้พิพากษา William ที่ว่าการจัดลำดับ(sequence) และการจัดวางองค์ประกอบ(organization) ที่เหมือนกันของ GEMS และ Telcot อาจจะเป็นผลมาจากข้อเท็จจริงเรื่องปัจจัยทางตลาดที่ว่าทั้งสองโปรแกรมที่ถูกออกแบบนั้นเป็นความคิดที่สามารถถูกแสดงออกได้ในวิธีทางเพียงวิธีเดียว และในคดีนี้เห็นว่า โครงสร้าง, ตรรกและผังที่เป็นส่วนสำคัญต่อความคิดของโปรแกรมไม่สามารถได้รับการคุ้มครองโดยกฎหมายลิขสิทธิ์

คดีต่อไปที่ศาลตัดสินว่าไม่ให้การคุ้มครองแก่ user interface คือ คดี Apple Computer Inc., v. Microsoft Corp. and Hewlett-Packard Company<sup>31</sup>

ในคดีนี้โจทก์ได้ฟ้องว่าจำเลยก๊อปปี้การแสดงที่ดึงดูด(fanciful display) และการปรากฏภาพ (images appearing) บนหน้าจอคอมพิวเตอร์ของ Macintosh การแสดงออกทางหน้าจอและภาพพจน์ที่แตกต่างที่มีในโปรแกรม Macintosh เป็นที่รู้จักอย่างแพร่หลายในฐานะสิ่งที่รับรองคุณภาพของระบบคอมพิวเตอร์ Macintosh ของ Apple ได้มีรูปภาพมากมายที่แสดงถึงการแสดงออกทางหน้าจอของ Macintosh กับ Window 2.03 ของ Microsoft

การแสดงออกทางหน้าจอที่มีการใช้รูปภาพที่ทำให้ง่ายและสะดวกในการใช้งาน ตัวอย่างเช่น อย่างส่วนของงานที่ไม่ต้องการก็เปรียบได้กับขยะ ซึ่งขยะ หรือ ของที่ไม่ต้องการใช้มันเป็นการง่ายที่จะใช้สัญลักษณ์รูปภาพหรือ icon ที่แตกต่างกัน เช่น ใช้หลุมสีดำ (black hole) หรือใช้รูปเครื่องย่อยกระดาษ จะไม่เป็นการละเมิดที่ก๊อปปี้ถึงขยะ (trash) ถ้าสิ่งที่ปรากฏบนหน้าจอแตกต่างกัน เป็นต้น<sup>32</sup> เพราะเรื่องของการใช้สัญลักษณ์แทนที่สิ่งของที่ไม่ต้องการใช้เป็นเพียงความคิดเท่านั้นที่กฎหมายลิขสิทธิ์ไม่คุ้มครอง ส่วนการใช้รูปอะไรแทนนั้นก็เป็นการแสดงออกซึ่งความคิด

Microsoft ได้รับอนุญาตจาก Apple เพื่อผลิต User interface ของโปรแกรม Window 1.0 แต่ในสัญญาอนุญาตไม่พบว่าให้สิทธิกับ Window รุ่นใหม่คือ Window รุ่น 2.03 และรุ่น 3.0 ที่มีลักษณะเป็นหน้าต่างซ้อนหน้าต่าง (overlapping window) และพัฒนาให้ icons สามารถเคลื่อนที่ไปตามจอภาพให้มีประสิทธิภาพมากขึ้น

ในเวลาอีกไม่กี่ปีได้มีผู้กล่าวว่า การใช้สัญลักษณ์รูป (icon manipulate) และ Macro จะล้ำสมัยในฐานะของ user interface และกว่าที่จะสร้างผู้ใช้ Lotus 1-2-3 ที่ชำนาญในการใช้โปรแกรมนี้ต้องใช้เงินจำนวนมากในการพัฒนาบุคคลากรนี้ นอกจากนั้นในปี 1980 เป็นโปรแกรมประยุกต์จำนวนเป็นแสนที่มีขายตามท้องตลาดถูกเขียนขึ้นสำหรับเครื่องคอมพิวเตอร์ส่วนบุคคล Apple II การให้การคุ้มครองแก่ Apple ก็จะทำให้เกิดอำนาจที่ผูกขาด และอำนาจที่

<sup>31</sup> 709 F. Supp. 925 (N.D. Cal.) 1989 cit 799 F. 2d 1006 (N.D. Cal.) 1992

<sup>32</sup> Anthony Lawrence Clapes, Software, copyright, and competition: The "look and feel" of the law, p. 199.

ผูกขาด(Monopoly power)เป็นอำนาจเพื่อกีดกันการแข่งขันและใช้เป็นเครื่องมือในการขึ้นราคาผลิตภัณฑ์<sup>33</sup>

โจทก์คือบริษัท Apple ได้ฟ้องจำเลยว่าจะเมิดลิขสิทธิ์ในส่วนที่เป็นการแสดงออกทางหน้าจอและได้ผิดสัญญาอนุญาตให้ใช้สิทธิ์นี้ โดยที่โปรแกรม Window 2.03 และ Window 3.0 ของบริษัท Microsoft เวลาใช้งานนั้นให้ความรู้สึกที่เหมือนกันการใช้โปรแกรมของโจทก์ ในการพิจารณาของศาลโดยผู้พิพากษา Walker ได้ปฏิเสธที่จะให้การคุ้มครองแก่การแสดงออกทางหน้าจอที่ทำให้ผู้ใช้เกิดความรู้สึกที่เหมือนกันที่เรียกว่า look and feel โดยที่ศาลเห็นว่าการออกแบบและรูปแบบการจัดหน้าจอนี้มีข้อจำกัดและเหตุผลในทางเทคนิค จึงทำให้สิ่งนี้ออกมาคล้ายคลึงกัน จึงเป็นสิ่งที่ไม่ได้รับการคุ้มครองตามกฎหมายลิขสิทธิ์

ในคดีนี้ได้มีนักกฎหมายท่านหนึ่งคือ Steven Schortgen ได้เสนอแนวทางในการให้การคุ้มครองแก่ Graphical User Interface(GUIs) โดยใช้หลักกฎหมายเรื่อง Trade Dress ใน Trademark Law ซึ่งกฎหมายเครื่องหมายการค้านี้ไม่เพียงแต่คุ้มครองถึงคำหรือวลีทางการค้าแต่ยังรวมถึงภาพวาด, โลโก้หรือรูปร่างที่มีลักษณะพิเศษของหีบห่อของผลิตภัณฑ์ และในเรื่องของ Trade Dress นั้นก็เป็นส่วนหนึ่งของกฎหมายเครื่องหมายการค้า Trade Dress นั้นหมายถึง รูปแบบที่ผู้ผลิตแสดงถึงผลิตภัณฑ์ของเขาต่อตลาด จึงเป็นสิ่งที่เกี่ยวข้องกับภาพพจน์ของผลิตภัณฑ์ทั้งหมดรวมถึงรูปลักษณ์ เช่น ขนาด, รูปร่าง, สี หรือการรวมกันของสี, ลักษณะพื้นผิว หรือรูปภาพ เป็นต้น นอกจากนี้การคุ้มครอง Trade Dress ยังมาจากมาตรา 43(a) ของกฎหมาย Lanham Act ที่ไม่จำกัดเฉพาะการละเมิดเครื่องหมายการค้าและ Trade Dress แต่ก่อนข้างกล่าวถึงรูปแบบทั้งหมดของการแข่งขันที่ไม่เป็นธรรม(unfair competition) เพื่อป้องกันไม่ให้เกิดการแข่งขันที่ไม่เป็นธรรม<sup>34</sup>

เมื่อเปรียบเทียบกับกฎหมายลิขสิทธิ์แล้ว Trade Dress เป็นกลไกที่ดีกว่าในการให้การคุ้มครองแก่ GUIs โดยมีเหตุผล 3 ประการคือ<sup>35</sup>

1. การเรียกร้องของ Trade Dress ที่ประสบความสำเร็จต้องการส่วนประกอบทางกฎหมายและทางข้อเท็จจริงที่แตกต่างจากสิ่งที่ต้องการของกฎหมายลิขสิทธิ์ เพื่อให้การฟ้องร้องมีส่วนประกอบที่ดีกว่าสำหรับการคุ้มครองซอฟต์แวร์

<sup>33</sup> Ibid., p. 205.

<sup>34</sup> Steven Schortgen, "Dressing up software interface protection: The application of two pesos to look and feel," *Cornell Law Review*, 80 (November 1994): 161-162.

<sup>35</sup> Ibid., p.160.

2. เนื่องจากประวัติศาสตร์ทางกฎหมายและพลังทางการจินตนาการ (animating forces) ระหว่าง Trade Dress และกฎหมายลิขสิทธิ์แตกต่างกันในสิ่งที่เป็นสาระสำคัญและวิธีการ

3. คำตัดสินของศาลสูงสุดของสหรัฐอเมริกาในคดี Two Pesos, Inc., V. Taco Cabana, Inc.,<sup>36</sup> สามารถใช้เป็นแนวทางสำหรับนักกฎหมายคอมพิวเตอร์ในการโต้แย้งว่าการคุ้มครอง Trade Dress สามารถปรับใช้ต่อสิ่งที่ไม่ใช่ตัวอักษร(non-literal) ของซอฟต์แวร์คอมพิวเตอร์และโดยเฉพาะเรื่อง look and feel ของโปรแกรมคอมพิวเตอร์

ในเรื่องของการคุ้มครอง Trade Dress นี้ได้มีคดีที่น่าสนใจคือคดี Two Pesos, Inc., V. Taco Cabana, Inc., ซึ่งเป็นเรื่องของความเหมือนกันของสองภัตตราคารนี้มาจากการปรากฏภาพทั้งหมดของร้านทั้งสองร้านมาจากการตกแต่ง, การทาสี, ผ้าใบบังแดดและประตูโรงรถ การมองเห็นในลักษณะของร้านอาหารนี้เป็นลักษณะที่เด่นถาวร ถือได้ว่าจำเลยได้ลอกเลียนแบบโจทก์ ดังนั้นการที่ศาลมองแบบนี้ นักกฎหมายทางคอมพิวเตอร์สามารถนำมาใช้เทียบเคียงกับการแสดงออกทางหน้าจอ(screen display) หรือส่วนประกอบอื่นๆ ของโปรแกรมคอมพิวเตอร์ ซึ่งเป็นสิ่งที่ประชาชนสามารถเห็นได้

เมื่อนำหลักการนี้มาปรับใช้กับคดี Apple V. Microsoft ดูเหมือนว่าจะเป็นประโยชน์ต่อโจทก์ ที่ฟ้องร้องว่า Microsoft ได้ลอกเลียนแบบในส่วนของการออกแบบหน้าจอ เพราะการที่ลักษณะของหน้าจอที่เหมือนกันนี้ประชาชนสามารถเห็นและรับรู้ได้ ในการพิจารณานั้นจะต้องพิจารณาถึงความสับสนของผู้บริโภคที่มีต่องานนั้นโดยสังเกตถึงการปรากฏของงานและภาพพจน์ทั้งหมดของผลิตภัณฑ์ Trade Dress ซึ่งสามารถนำไปปรับใช้กับเรื่องการให้การคุ้มครอง look and feel ของซอฟต์แวร์ได้ เนื่องจาก Apple ไม่ได้อ้างถึงเรื่อง Trade Dress แต่อ้างเฉพาะเรื่องการฟ้องละเมิดลิขสิทธิ์จึงทำให้ Microsoft ประสบความสำเร็จในการต่อสู้คดีที่ว่ากระทำของตนไม่เป็นการละเมิดลิขสิทธิ์

คดีสุดท้ายที่จะกล่าวถึงเป็นเรื่องแนวทางที่ศาลไม่ให้การคุ้มครองแก่ User interface คดี Lotus Development Corp., V. Borland Internation Inc.,<sup>37</sup>

คดีนี้ศาลได้จำกัดขอบเขตของการคุ้มครองลิขสิทธิ์สำหรับโปรแกรมคอมพิวเตอร์ถึงแม้ว่าลำดับสูงต่ำของเมนูคำสั่งของ Lotus (Lotus menu command hierarchy) จะมีการแสดงออก

<sup>36</sup> 112 S. Ct. 2753, reh'g denied, 113 S.Ct. 20 (1992)

<sup>37</sup> 788 F. Supp. 78 (D. Mass.) 1992



รวมอยู่แต่มันไม่สามารถเป็นงานอันมีลิขสิทธิ์ได้เพราะว่ามันเป็นส่วนหนึ่งของวิธีการของการปฏิบัติการ (method of operation) ซึ่งถูกกีดกันจากการคุ้มครองลิขสิทธิ์ตามหลักที่มีในมาตรา 102(b)<sup>38</sup>

Lotus และ Borland ได้โต้แย้งกันเกี่ยวข้องกับ electronic spreadsheets ที่ Lotus ขายโปรแกรมภายใต้ชื่อ Lotus 1-2-3 และ Borland ขายโปรแกรมตามมากภายใต้ชื่อ Quattro โปรแกรม Lotus 1-2-3 จะมีลักษณะเป็นตารางของแถวและคอลัมน์โดยเป็นตัวอักษรด้านคอลัมน์และเป็นตัวเลขด้านแถว โปรแกรม Lotus 1-2-3 ผู้ใช้สามารถสร้าง Macro language ของเขาเอง ในบางส่วนของโปรแกรม Quattro ได้บรรจุการติดต่อเชื่อมประสานที่เลียนแบบ Lotus (Lotus Emulation Interface) ซึ่งสามารถทำให้ผู้ใช้ Quattro รู้สึกราวกับว่าได้ใช้โปรแกรม Lotus 1-2-3 อยู่ การใช้งานได้ (compatibility) บางส่วนกับโปรแกรม Lotus 1-2-3 ที่ประสบความสำเร็จของ Borland โดยผ่าน Quattro Key Reader ความเหมือนกันนี้ทำให้ผู้ใช้ Macro ของโปรแกรม Lotus 1-2-3 สามารถใช้ Macro ของ Quattro ได้

ในคดีนี้ศาลอุทธรณ์ได้ตัดสินเหมือนศาลชั้นต้นที่ว่า menu command hierarchy ของ Lotus เป็นวิธีการของการปฏิบัติ (a method of operation) จึงไม่ได้รับการคุ้มครองโดยกฎหมายลิขสิทธิ์ คำตัดสินของศาลอุทธรณ์ในประเด็นนี้มีเหตุผลน่าเชื่อถือ แต่มีเหตุผลหนึ่งที่ไม่ได้รับความเชื่อถือ คือการที่ศาลใช้การเทียบเคียงกับปุ่มของ VCR เช่น ปุ่ม Record, Play, Reverse กับ menu command hierarchy ของโปรแกรม Lotus ลักษณะของปุ่ม VCR เป็นวิธีการของการปฏิบัติ (method of operation) ศาลเทียบเคียงโดยชี้ว่า Lotus menu command hierarchy ไม่เท่ากับป้าย (labels) บนปุ่มของ VCR แต่ให้เท่ากับปุ่มในตัวของมันเอง ได้มีข้อโต้แย้งถึงการเทียบเคียงของศาลว่าไม่เหมาะสม คือ โปรแกรมคอมพิวเตอร์ได้รับการปฏิบัติในฐานะงานวรรณกรรม และการวิเคราะห์ของศาลละเลยในแง่ที่เกี่ยวข้องกับการแสดงออกของ menu command hierarchy สิ่งนี้สามารถถูกแสดงให้เห็นโดยจำนวนที่มากของความหลายที่เป็นไปได้ ไม่เหมือนกับปุ่มบน VCR แต่ Lotus ไม่ได้พยายามเพื่อการคุ้มครองปุ่ม (button) ของโปรแกรม แต่เป็น

---

<sup>38</sup> 17 U.S.C. section 102(b) "In no case does copyright protection for an original work of authorship extend to any idea, procedure, process, system, method of operation, concept, principle, or discovery, regardless of the form in which it is described, explained, illustrated, or embodied in such work."

การพยายามเพื่อคุ้มครองถึงการแสดงออกของปุ่มในฐานะที่รวมของเมนูคำสั่ง(menu commands) และ ลำดับสูงต่ำของเมนูคำสั่ง(menu command hierarchy)<sup>39</sup>

ศาลอุทธรณ์ที่ 1 ได้ปฏิเสธการทดสอบ Abstraction-Filtration-Comparison ที่ใช้โดยผู้พิพากษา Keeton ที่เป็นผู้พิพากษาศาลชั้นต้น ศาลได้ให้เหตุผลที่ว่า ที่ไม่นำมาปรับใช้เพราะว่ากรณีนี้เกี่ยวข้องกับกรอปปี้ตัวอักษร ของ Lotus menu command hierarchy มากกว่าคดี Altai ที่เป็นการลอกเลียนในลักษณะที่ไม่เป็นตัวอักษรของรหัสคอมพิวเตอร์ แต่จริงแล้วการทดสอบ Abstraction-Filtration-Comparison สามารถนำมาปรับใช้กับคดีได้ทุกประเภทดังที่ศาลในคดีต่างๆ ได้นำมาปรับใช้

นอกจากนี้คดี Baker V. Selden ยังสามารถนำมาปรับใช้กับคดี Lotus V. Borland รูปแบบของ Selden เป็นการ interface ระหว่างระบบของ Selden และผู้ใช้ ระบบบัญชีการเก็บหนังสือของ Selden เป็นการพัฒนาที่เพิ่มมากขึ้นโดยอาศัยระบบก่อนๆ ของบัญชี Lotus โดยอาศัยระบบ electronic spreadsheet ที่มีมาก่อนเหมือนกับ Selden โดยการที่ Lotus เริ่มต้นด้วยเทคโนโลยีที่มีอยู่ก่อน และสร้างระบบสำหรับใส่ข้อมูลลงในคอมพิวเตอร์, การจัดการข้อมูล, การแสดงข้อมูลบนหน้าจอ และการใส่ผลที่ได้รับลงในกรอปปี้<sup>40</sup> เป็นวิทยาการที่ก้าวหน้า สลับซับซ้อนมากกว่าระบบบัญชีของ Selden แต่มันไม่ใช่ข้อแตกต่างที่สำคัญระหว่างสิ่งนี้กับกฎหมายลิขสิทธิ์ที่เป็นประเพณีที่ใช้พิจารณา ยิ่งไปกว่านั้นโครงสร้างเมนูของระบบ Lotus เป็นการจลลำดับของการปฏิบัติการโดยเปรียบเทียบต่อ ขบวนการสั่ง (ordered process)ของการเก็บหนังสือที่ถูกอธิบายในหนังสือของ Selden ดังนั้นเมื่อระบบบัญชีการเก็บหนังสือของ Baker ยังไม่ได้รับการคุ้มครอง การติดต่อเชื่อมประสานระหว่างเครื่องคอมพิวเตอร์และผู้ใช้ของ Lotus 1-2-3 นำที่จะไม่ได้รับการคุ้มครองเช่นเดียวกัน

จากคดีต่างๆ ที่ได้กล่าวมาแล้วอาจสรุปได้ว่าแนวของศาลสหรัฐอเมริกามีแนวโน้มไปในทางที่จะไม่ให้การคุ้มครองแก่การติดต่อประสานระหว่างเครื่องคอมพิวเตอร์กับผู้ใช้(user interface) เพราะศาลเห็นว่าถ้าให้การคุ้มครองสิ่งนี้เท่ากับว่าเป็นการให้สิทธิผูกขาดแก่ผู้ผลิต

<sup>39</sup> Bradford E. Kile, Ruffin B. Cordell and Michael J. McKeon, "Lotus V. Borland: Copyright protection of computer software in a state of transition," *Copyright World* 53 (September 1995): 20.

<sup>40</sup> Dennis S. Karjala and Peter S. Menell, "Applying fundamental copyright principles to Lotus Development Corp. V. Borland International, Inc.," *High Technology Law Journal* 10(January 1995): 187.

ขึ้นมาเป็นคนแรกและจะเป็นอุปสรรคแก่คู่แข่งที่จะเข้าสู่ตลาดตามมาโดยที่เขาจะไม่สามารถพัฒนาโปรแกรมนั้นต่อไปได้ ซึ่งจะเป็นผลเสียต่อประชาชนและสังคมส่วนรวม

## ข. กลุ่มประชาคมยุโรป

ในข้อบังคับของ EC ได้บัญญัติไว้อย่างชัดเจนในมาตรา 1(2) ว่า “Ideas and principles which underlie any element of a computer program, including those which underlie its interfaces, are not protected by copyright under this Directive.” ความหมายของ “Interface” เป็นสิ่งที่ยากในการที่จะให้คำจำกัดความ แต่ในข้อบังคับนี้ได้ให้คำนิยามของคำนี้ไว้ในคำนำที่ 11<sup>41</sup> ที่อธิบายว่าเป็นส่วนหนึ่งของโปรแกรมคอมพิวเตอร์ซึ่งเตรียมสำหรับการติดต่อระหว่างกันและปฏิกิริยาตอบโต้ระหว่างส่วนของซอฟต์แวร์และฮาร์ดแวร์ที่โดยทั่วไปรู้จักในนาม interface ปัญหาในเรื่องของ interface นี้เป็นผลมาจากลักษณะทางธรรมชาติที่พิเศษของโปรแกรมคอมพิวเตอร์ที่ไม่เพียงแต่จะแตกต่างจากงานวรรณกรรมทั่วไป แต่ยังแตกต่างจากงานที่ปฏิบัติงานอื่นๆ (other functional work) ที่ได้รับการคุ้มครองภายใต้กฎหมายลิขสิทธิ์ ลักษณะของโปรแกรมคอมพิวเตอร์ที่ประกอบด้วยส่วนต่างๆที่เป็นระบบปฏิบัติการข้อมูลที่สมบูรณ์ (complete data-processing system) ดังนั้นจึงจำเป็นต้องมีการการติดต่อด้วยส่วนต่างๆ ของโปรแกรมที่เป็นทั้งส่วนของฮาร์ดแวร์และซอฟต์แวร์<sup>42</sup>

จะเห็นได้ว่าเรื่องของการติดต่อเชื่อมประสาน(interface) นี้มีความสำคัญมากโดยเป็นจุดสำคัญในการแข่งขันเพราะถ้าผู้ผลิตสามารถตอบสนองความต้องการของผู้ใช้โดยมี interface ที่ให้ความสะดวกแก่ผู้ใช้โปรแกรมคอมพิวเตอร์นั้นย่อมได้รับส่วนแบ่งในตลาดที่สูง และถ้าเขาสามารถผูกขาดการใช้ interface ของเขาโดยใช้ข้อบังคับของ EC นี้ย่อมทำให้คู่แข่งไม่สามารถพัฒนาโปรแกรมเข้าสู่ท้องตลาดได้เนื่องจากกลัวว่าเป็นการละเมิดลิขสิทธิ์จึงเป็นเหตุผลหนึ่งที่ข้อบังคับของ EC นี้ไม่ให้การคุ้มครองแก่ Interface โดยได้บัญญัติไว้อย่างชัดเจน และหลักการที่ไม่ให้การคุ้มครองแก่การติดต่อเชื่อมประสานนั้นเป็นหลักการที่ได้รับการยอมรับจากประเทศสมาชิกประชาคมยุโรป เช่น อังกฤษ, เยอรมัน เป็นต้น

<sup>41</sup> EC Directive on the legal protection of computer programs Recital 11 “Whereas the parts of the program which provide for such interconnection and interaction between elements of software and hardware are generally known as ‘interfaces;’”

<sup>42</sup> Michael Sucker, “The Software Directive—Between the combat against piracy and the preservation of undistorted competition,” p. 14. ใน A Handbook of European Software Law, M. Lehmann and C.F. Tapper eds. (London: Clarendon Press., 1993)

## ค. ประเทศไทย

ตามพระราชบัญญัติลิขสิทธิ์ พ.ศ. 2537 ไม่มีส่วนไหนของตัวบทที่ได้กล่าวถึง การติดต่อเชื่อมประสาน(Interface) เลยว่าหมายถึงอะไร และจะได้รับการคุ้มครองโดยกฎหมายลิขสิทธิ์หรือไม่ รวมทั้งยังไม่มีแนวคำพิพากษาที่ตัดสินในเรื่องนี้เลย ผู้เขียนเห็นว่ากฎหมายลิขสิทธิ์ของไทยน่าจะบัญญัติถึงเรื่องนี้ให้ละเอียดเหมือนอย่างข้อบังคับของ EC ว่า Interface หมายถึงอะไรและจะได้รับการคุ้มครองหรือไม่อย่างไร แต่จากตัวบทที่มีอยู่นั้นผู้เขียนเห็นว่าสามารถปรับเรื่องนี้ให้เข้ากับหลักเกณฑ์ที่มีในมาตรา 6 วรรคสองได้ โดยให้ถือว่า Interface เป็นระบบหรือวิธีใช้หรืองานที่กฎหมายลิขสิทธิ์ไม่ให้การคุ้มครองเนื่องจากลักษณะของ Interface เป็นเรื่องของการติดต่อและการทำงานของโปรแกรมคอมพิวเตอร์

นอกจากนั้นผู้เขียนยังเห็นว่าการติดต่อเชื่อมประสาน(Interface)เป็นส่วนของโปรแกรมคอมพิวเตอร์ที่ไม่สมควรได้รับการคุ้มครองเพราะ เนื่องจากส่วนหนึ่งเป็นความจำเป็นที่จะต้องมีการเชื่อมประสานกันหรือต้องมีการทำงานร่วมกันในระหว่างโปรแกรมคอมพิวเตอร์ด้วยกันเองหรือทำงานร่วมกับฮาร์ดแวร์ การที่ต้องมีการทำงานเชื่อมติดต่อประสานกันนี้จะเป็นข้อจำกัดทางหนึ่งในการเขียนโปรแกรมคอมพิวเตอร์ที่ทำให้โปรแกรมคอมพิวเตอร์ในบางส่วนมีความเหมือนคล้ายกัน ซึ่งเป็นลักษณะของการที่ต้องมีการติดต่อเชื่อมประสานนี้ไม่ว่าจะเป็นระหว่างผู้ใช้กับโปรแกรมคอมพิวเตอร์หรือระหว่างโปรแกรมคอมพิวเตอร์กับเครื่องคอมพิวเตอร์หรือในระหว่างโปรแกรมคอมพิวเตอร์ด้วยกันเองทำให้งานโปรแกรมคอมพิวเตอร์แตกต่างกันไปจากงานวรรณกรรมโดยทั่วไปที่ไม่มีข้อจำกัดอย่างนี้ และถ้าให้การคุ้มครองแก่ Interface นั้นจะทำให้ผู้ที่เข้ามาสู่ตลาดภายหลังไม่สามารถสร้างโปรแกรมเข้ามาแข่งขันได้ เพราะผู้ใช้ส่วนมากมักเคยชินกับ การใช้เครื่องที่มี user Interface ที่ตนเองคุ้นเคยอยู่ และไม่เปิดใจรับกับโปรแกรมคอมพิวเตอร์ใหม่ๆ ที่มีการ Interface ที่ตัวเองไม่มีความคุ้นเคย และถ้าจะใช้โปรแกรมใหม่ก็จำเป็นที่จะต้องไปศึกษาเรียนรู้อีกซึ่งจะเสียทั้งเวลาและค่าใช้จ่ายในการพัฒนาบุคลากรขึ้นมา