

บทที่ 2

สรุปสาระสำคัญจากเอกสารที่เกี่ยวข้อง

2.1 อาร์ไอเอ (RIA : Rich Internet applications)

อาร์ไอเอคือลักษณะการทำงานของแอปพลิเคชัน ที่ส่วนลูกข่ายจะทำหน้าที่เป็นส่วนตอบโต้กับผู้ใช้ซึ่งโค้ดโปรแกรมของฝั่งเครื่องลูกข่ายจะทำหน้าที่เพียงตอบโต้การกระทำของผู้ใช้ ทำให้มีความสามารถคล้ายกับแอปพลิเคชันบนเครื่องลูกข่ายเอง แต่การทำงานหลักจะถูกประมวลผลบนฝั่งเครื่องแม่ข่าย อาร์ไอเอมักจะถูกนำไปใช้ในเว็บแอปพลิเคชัน ซึ่งลักษณะการทำงานของเอเจ็ทซ์เป็นลักษณะการทำงานแบบอาร์ไอเออย่างหนึ่ง

อาร์ไอเอถูกแนะนำขึ้นครั้งแรกโดยมาโครมีเดีย (Macromedia) ในปี ค.ศ.2002 มีจุดประสงค์เพื่อสร้างเว็บแอปพลิเคชันให้สามารถตอบโต้และใช้งานได้เสมือนแอปพลิเคชันบนเครื่องลูกข่ายเอง แต่ยังคงรักษาสถานะและข้อมูลไว้ที่เครื่องคอมพิวเตอร์แม่ข่าย

มีเครื่องมือหลายเครื่องมือที่ถูกพัฒนามาเพื่อใช้กับอาร์ไอเอ ซึ่งแยกได้เป็นสองประเภทคือประเภทที่ต้องติดตั้งซอฟต์แวร์เพิ่มเติม และประเภทที่ไม่ต้องติดตั้งซอฟต์แวร์เพิ่มเติมแต่อาศัยความสามารถที่มีอยู่ของเบราว์เซอร์ในการทำงาน เครื่องมือที่ถูกนำมาใช้ที่รู้จักกันอย่างแพร่หลายได้แก่

- JavaScript/Ajax
- Adobe Flash, Adobe Flex and Adobe AIR
- Java applets
- ActiveX Controls

2.2 จาวาสคริปต์ (JavaScript)

จาวาสคริปต์เป็นภาษาแบบสคริปต์ถูกพัฒนาขึ้นในปี ค.ศ.1995 และถูกใช้ในส่วนลูกข่ายของการพัฒนาเว็บแอปพลิเคชัน รูปแบบภาษาจาวาสคริปต์เป็นไปตามมาตรฐาน ECMAScript ซึ่งได้รับอิทธิพลมาจากภาษา ซี และจาวา จาวาสคริปต์ถูกนำมาใช้ในเว็บเบราว์เซอร์เกือบทุกเบราว์เซอร์จนถึงได้ว่าเป็นส่วนหนึ่งของเบราว์เซอร์ ทำให้เบราว์เซอร์สามารถตอบสนองการทำงานและโต้ตอบกับผู้ใช้ได้อย่างมีประสิทธิภาพ

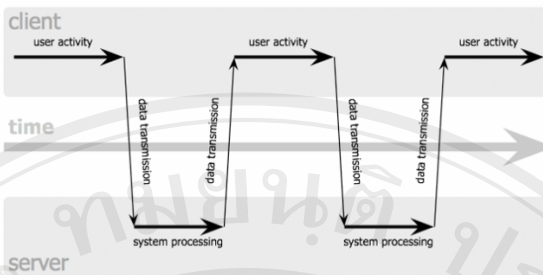
2.3 เอแจ็กซ์ (Ajax : Asynchronous JavaScript And XML)

เอแจ็กซ์เป็นการรวมเทคโนโลยีจากความสามารถของจาวาสคริปต์ ถูกนำเสนอครั้งแรกในปี ค.ศ. 2005 แต่โดยวิธีการและลักษณะการทำงานนั้นได้เกิดขึ้นนานแล้ว โดยบริษัทไมโครซอฟต์ (Microsoft) เริ่มนำการเรียกหน้าแสดงผลของเว็บแยกออกมาจากหน้าแสดงผลหลัก โดยใช้แท็กไอเฟรม (IFRAME Tag) ของโครงสร้างเอชทีเอ็มแอลในการจัดการ เพื่อพัฒนาการทำงานในแบบรีโมทสคริปต์ (Remote Script) ต่อมาจึงได้สร้างออบเจกต์ (object) ที่สามารถส่งคำขอและรอรับข้อความจากเครื่องแม่ข่ายเพื่อมาจัดการแทนแท็กไอเฟรมและเรียกออบเจกต์นั้นว่า เอแจ็กซ์เอ็มแอลเอชทีทีพีรีเควสท์ (XMLHttpRequest) และจากเทคนิคดังกล่าวนี้ก็ถูกพัฒนาความสามารถด้านต่างๆเพิ่มเติมจนสรุปมาเป็นเอแจ็กซ์ ซึ่งเอแจ็กซ์ประกอบด้วยเทคนิคที่สำคัญดังต่อไปนี้

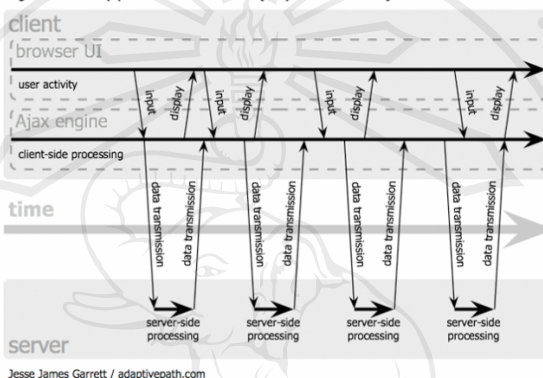
- 0) เอชทีเอ็มแอล (HTML : Hyper Text Markup Language) และ ซีเอสเอส (CSS : Cascading Style Sheets)
- 2) ดีโอเอ็ม (DOM : Document Object Model) สำหรับจาวาสคริปต์
- 0) เอแจ็กซ์เอ็มแอลเอชทีทีพีรีเควสท์ (XMLHttpRequest)
- 0) เอแจ็กซ์เอ็มแอล (XML : Extensible Markup Language) สำหรับจาวาสคริปต์

ลักษณะการทำงานของเอแจ็กซ์และเว็บแอปพลิเคชันทั่วไปแสดงเป็นช่วงเวลาในการติดต่อกับเครื่องแม่ข่ายได้ดังรูปที่ โดยสังเกตจากช่วงเวลาทำงานบนฝั่งเครื่องลูกข่าย เว็บแอปพลิเคชันทั่วไปจะเสียเวลาในการรอรับข้อมูลกลับมาจากเครื่องแม่ข่าย ส่วนเอแจ็กซ์จะยังสามารถทำงานต่อไปได้เพราะเอแจ็กซ์จะทำหน้าที่คอยรับข้อมูลแทนเว็บเบราว์เซอร์

classic web application model (synchronous)



Ajax web application model (asynchronous)



รูปที่ 2.1 ช่วงเวลาในการติดต่อกับเครื่องแม่ข่ายของเว็บแอปพลิเคชันทั่วไปกับเอแจ็กซ์

2.4 เอชทีเอ็มแอล (HTML : Hyper Text Makeup Language) และ ซีเอสเอส (CSS : Cascading Style Sheets)

เอชทีเอ็มแอลเป็นเอกสารที่ใช้แสดงข้อมูลผ่านเครือข่ายอินเทอร์เน็ต ถูกนำเสนอในปี ค.ศ. 1991 และถูกกำหนดให้มีมาตรฐานในปี 1997 โดยดับเบิลยูทีซี (W3C : World Wide Web Consortium) รูปแบบโครงสร้างของเอกสารประกอบขึ้นมาจากแท็ก (Tag) และถูกอ่านและแปลงออกมาเป็นการแสดงผลโดยโปรแกรมเรียกว่าบราวเซอร์ (Browser) ต่อมาจึงเป็นเอกสารหลักที่ใช้ในการพัฒนาเว็บแอปพลิเคชัน จากความแพร่หลายของการใช้เอกสารบนเครือข่ายอินเทอร์เน็ตทำให้มีการเพิ่มความสามารถของเอชทีเอ็มแอลขึ้น โดยมีการเพิ่มความสามารถในการแสดงผลโดยซีเอสเอส

ซีเอสเอสเป็นการกำหนดรูปแบบการแสดงผลให้กับแท็กของเอชทีเอ็มแอล ในด้าน สี รูปแบบตัวอักษร และตำแหน่งของแท็ก เป็นต้น จากความสามารถของซีเอสเอสทำให้หน้าแสดงผลของเว็บแอปพลิเคชันมีความหลากหลายและยืดหยุ่นมากขึ้น

และเมื่อมีการใช้ภาษาสคริปต์บนบราวเซอร์ ภาษาสคริปต์สามารถจัดการกับเอชทีเอ็มแอล และซีเอสเอส ผ่านทางดีไอเอ็ม (DOM)

2.5 ดีโอเอ็ม (DOM : Document Object Model)

ดีโอเอ็มเป็นการจัดโครงสร้างของเอกสารให้อยู่ในรูปออบเจ็กต์ เพื่อให้โปรแกรมสามารถควบคุม ลด หรือเพิ่มเติม โครงสร้างของเอกสารได้สะดวก ดีโอเอ็มถูกกำหนดรูปแบบให้เป็นมาตรฐานโดยดับเบิลยูทีซี (W3C) เพื่อให้สามารถใช้คู่กับเอชทีเอ็มแอลได้อย่างมีประสิทธิภาพ และจาวาสคริปต์สามารถจัดกับเอชทีเอ็มแอลผ่าน โครงสร้างของดีโอเอ็ม

2.6 เอ็กซ์เอ็มแอล (XML : Extensible Markup Language)

เอ็กซ์เอ็มแอลเป็นเอกสารที่ใช้เก็บข้อมูลเพื่อนำไปประมวลผลต่อ โดยโครงสร้างของเอ็กซ์เอ็มแอลได้รับอิทธิพลมาจากเอชทีเอ็มแอลแต่มีกฎของโครงสร้างที่ตายตัวและชัดเจนกว่าเอชทีเอ็มแอล ซึ่งต่อมาเอชทีเอ็มแอลได้นิยมเขียนตามกฎและโครงสร้างของเอ็กซ์เอ็มแอลด้วย เอ็กซ์เอ็มแอลประกอบขึ้นจากแท็ก (Tag) เพื่อบอกความหมายของข้อมูลแต่ละส่วนและภายในแท็กอาจจะเป็นส่วนหนึ่งของข้อมูลหรือแท็กอื่นได้ จากโครงสร้างที่ยืดหยุ่นของเอ็กซ์เอ็มแอลทำให้กลายเป็นรูปแบบเอกสารที่นิยมสำหรับส่งข้อมูลผ่านเครือข่ายอินเทอร์เน็ต

จาวาสคริปต์สามารถจัดการกับเอกสารเอ็กซ์เอ็มแอลผ่านทางโครงสร้างดีโอเอ็ม (DOM)

2.7 เอ็กซ์เอสแอลที (XSLT : Extensible Stylesheet Language Transformations)

เอ็กซ์เอสแอลทีเป็นเอกสารในรูปแบบเอ็กซ์เอ็มแอล (XML) รูปแบบหนึ่ง แต่เป็นเอกสารที่มีความหมายในเชิงการแปลงข้อมูลในเอ็กซ์เอ็มแอลไปเป็นเอกสารรูปแบบอื่นรวมถึงรูปแบบเอ็กซ์เอ็มแอลหรือเอชทีเอ็มแอลด้วย โดยเอ็กซ์เอสแอลทีจะมีโครงสร้างแท็ก (Tag) ที่เป็นมาตรฐานเฉพาะเพื่อให้ความหมายในการแปลงข้อมูลเอ็กซ์เอ็มแอล

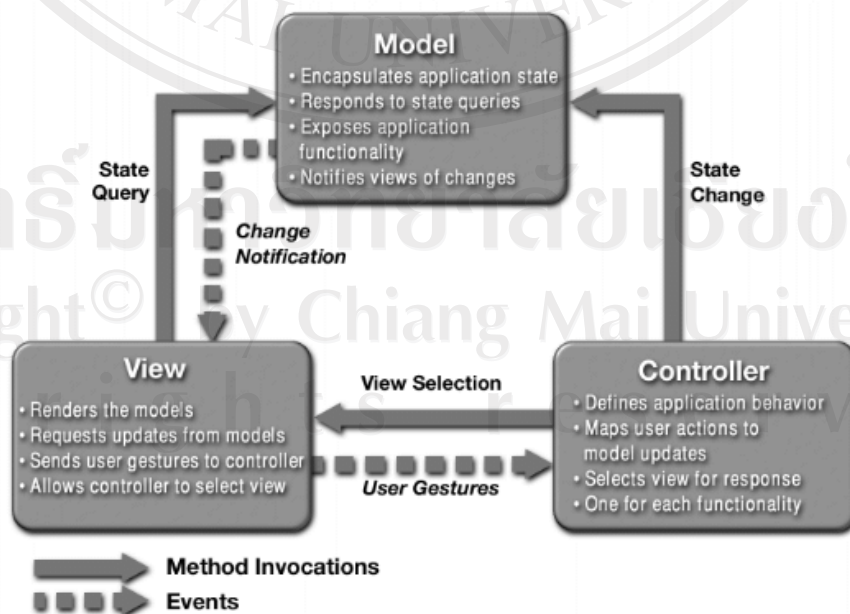
เอ็กซ์เอสแอลทีมิได้เป็นเทคนิคส่วนหนึ่งของเอเจ็ทซ์โดยตรง แต่เนื่องจากบราวเซอร์ส่วนใหญ่สามารถรองรับการทำงานของเอ็กซ์เอสแอลทีและความสะดวกในการใช้งานคู่กับเอ็กซ์เอ็มแอล จึงทำให้มักมีการใช้เอ็กซ์เอสแอลทีในการพัฒนาเอเจ็ทซ์

2.8 เอ็มวีซี (MVC : Model View and Controller)

เอ็มวีซีคือรูปแบบการออกแบบที่จัดแบ่งส่วนการทำงานออกเป็นกลุ่มๆตามลักษณะงานที่ต้องทำ โดยแบ่งงานเป็น 3 กลุ่มดังนี้

- 1) ส่วนปรับปรุงและติดต่อกับข้อมูล จะสร้างการทำงานส่วนนี้ออกมาเป็นหุ่นจำลอง (model) ซึ่งจะมีลักษณะการทำงานที่เหมือนกันไม่ว่าข้อมูลจะมีลักษณะใด ทำให้ส่วนที่จะนำโมเดลไปใช้เข้าใจเพียงตัวโมเดลก็เพียงพอไม่จำเป็นต้องเข้าใจข้อมูลที่โมเดลจัดการอยู่
- 4) ส่วนสร้างรูปแบบจากโมเดล จะนำโมเดลมาแปลงเป็นข้อมูลที่ผู้ใช้สามารถเข้าใจได้ (view) ซึ่งโมเดลหนึ่งตัวสามารถแปลงเป็นวิวได้หลายแบบ และวิวแต่ละแบบก็สามารถรองรับโมเดลได้หลายตัว
- 4) ส่วนทำหน้าที่จัดการคำขอและทำการเลือก โมเดลหรือวิวให้แก่ผู้ใช้ (controller) เมื่อผู้ใช้ตัดสินใจส่งคำขอมายังคอนโทรลเลอร์ คอนโทรลเลอร์จะทำการเลือกหรือปรับปรุงโมเดลตามคำขอ และจะทำการเลือกวิวเพื่อแปลงโมเดลส่งกลับไปให้ผู้ใช้ โดยมากแล้วคอนโทรลเลอร์หนึ่งตัวจะทำงานเพียงงานเดียว เช่น คอนโทรลเลอร์ปรับปรุงข้อมูล หรือคอนโทรลเลอร์แสดงรายการข้อมูล เป็นต้น

เอ็มวีซีเป็นรูปแบบการออกแบบที่แบ่งส่วนของการประมวลผลระหว่างข้อมูลหรือ โมเดล (Model) ออกจากการแสดงผลหรือวิว (View) เพื่อลดความซ้ำซ้อนของการพัฒนาและสามารถดูแลรักษาได้ง่าย จากลักษณะการทำงานนี้ทำให้ลดความซ้ำซ้อนในการสร้างส่วนอ่านข้อมูลลง เพราะข้อมูลจะถูกสร้างเป็นโมเดลซึ่งสามารถส่งไปยังวิวต่างๆที่เลือกได้ ลักษณะการทำงานของเอ็มวีซีแสดงดัง รูปที่



รูปที่ 2.2 แผนผังการทำงานของเอ็มวีซี

2.9 เทคนิคการทำงานของจาวาสคริปต์เฟรมเวิร์คสำหรับเอแจ็กซ์เทคโนโลยี ที่ทำการศึกษา

จากการศึกษาจาวาสคริปต์เฟรมเวิร์ค ได้แก่ เอ็กซ์เจเอส (Ext js) โพรโตไทป์ (Prototype) และโดโจ (Dojo) และในส่วนของเอ็มวีซี ได้ข้อสรุปเป็นเทคนิคที่น่าสนใจและลักษณะที่สามารถนำมาประยุกต์ใช้ได้ดังนี้

2.9.1 เอ็กซ์เจเอส (Ext js)

เอ็กซ์เจเอสเป็นจาวาสคริปต์เฟรมเวิร์คที่ถูกนำไปประยุกต์ใช้มากที่สุดตัวหนึ่ง เนื่องจากได้เตรียมชิ้นส่วนที่ใช้ติดต่อกับผู้ใช้ (UI widgets) ได้อย่างมากมาย ซึ่งผู้พัฒนาสามารถเรียกใช้ได้โดยไม่ต้องจัดการเพิ่มเติม

เทคนิคที่น่าสนใจของเอ็กซ์เจเอสคือ การสร้างขอบเขตการทำงาน (scope) โดยใช้รูปแบบโค้ดของฟังก์ชัน (function syntax) รวมไปถึงการส่งพารามิเตอร์ที่เป็นฟังก์ชันด้วย เช่น

```
var readData = (function(){
    var return_func = null
    if('text' == dataType) return_func = function(){
        alert(data);
    };
    if('binary' == dataType) return_func = function(){
        alertBinary(data);
    };
    return return_func;
})();
```

จากตัวอย่างตัวแปร readData ถูกกำหนดค่าโดยการใช้ขอบเขตการทำงาน ซึ่งในขอบเขตการทำงานนี้จะทำการเลือกฟังก์ชันที่จะคืนกลับไปให้ตัวแปร readData ใช้งาน โดยตัวแปร return_func จะถูกใช้งานภายในขอบเขตของฟังก์ชันเท่านั้น ซึ่งในงานวิจัยนี้ได้ใช้กระบวนการนี้เป็นกระบวนการหลักในการประกาศค่าให้กับเอ็กซ์เทนชันของเฟรมเวิร์ค

2.9.2 โพรโตไทป์ (Prototype)

โพรโตไทป์เป็นจาวาสคริปต์เฟรมเวิร์คที่ถูกใช้เป็นเฟรมเวิร์คหลักของรูบี้ออนเรล (Ruby on Rail) ซึ่งได้เตรียมความสามารถด้านเอแจ็กซ์และดีไอเอ็มเป็นหลัก โดยเตรียมฟังก์ชัน (Function) (\$) ไว้ เพื่อให้ผู้พัฒนาสามารถเรียกดีไอเอ็มผ่านทางไอดี โดยฟังก์ชันนี้มาสามารถ

ทำงานได้บนทุกเบราว์เซอร์ (cross-browser) ซึ่งดีไอเอ็มที่ได้นี้จะถูกเพิ่มเติมกระบวนการของเฟรมเวิร์คเข้าไปผ่านทางโปรโตไทป์ของดีไอเอ็ม และแต่ละกระบวนการที่เพิ่มเติมเข้าไปจะส่งวัตถุที่มันกระทำกระบวนการกลับมา เพื่อให้ผู้พัฒนาสามารถเรียกกระบวนการอื่นต่อไปได้ เช่น

```
$('#element_id').addClassName('comment').show();
```

จากตัวอย่างจะเป็นการเรียกดีไอเอ็มที่มีไอดีเป็น element_id มาเพิ่มเติมกระบวนการของเฟรมเวิร์ค และทำการเรียกเมธอด addClassName ซึ่งกระบวนการนี้จะส่งดีไอเอ็มเดิมกลับมาเพื่อเรียกใช้เมธอด show ต่อไป ในส่วนของเอเจ็ทซ์ได้เตรียมฟังก์ชันให้เรียกใช้ซึ่งสามารถทำงานได้บนทุกเบราว์เซอร์ แต่ฟังก์ชันที่เตรียมให้ไม่ได้มีความสัมพันธ์โดยตรงกับดีไอเอ็ม

จุดเด่นที่สำคัญของโปรโตไทป์และถูกนำมาประยุกต์ใช้กับงานวิจัยคือ มีการต่อเติมเมธอดของฟังก์ชันมาตรฐานผ่านทาง โปรโตไทป์ ซึ่งวัตถุที่สร้างจากฟังก์ชันนี้จะสามารถเรียกใช้เมธอดของเฟรมเวิร์คร่วมกันได้

2.9.3 โดโจ (Dojo)

โดโจเป็นจาวาสคริปต์เฟรมเวิร์คขนาดใหญ่ ซึ่งเตรียมกระบวนการในด้านต่างๆ ไว้อย่างมากมาย ทั้งในด้านเหตุการณ์ ลักษณะการเคลื่อนไหว (Animation) และอื่นๆ ซึ่งวิธีการที่สำคัญของโดโจคือ ความสามารถในการอ่านเอชทีเอ็มแอลแท็กที่มีรูปแบบตามที่โดโจกำหนดไว้ จากวิธีการนี้ผู้พัฒนาไม่จำเป็นต้องพัฒนาในส่วนจาวาสคริปต์แต่เพียงอย่างเดียว แต่สามารถแทรกเอชทีเอ็มแอลแท็กเพื่อให้การกระทำของจาวาสคริปต์มีผลในกรณีต่างๆ ได้ เช่น

```
<div dojoType="dijit.Declaration"
    widgetClass="TextField" defaults="{defaultText:}">
  <input type="text" dojoAttachEvent="onchange"
    dojoAttachPoint="tNode" value="{defaultText}"/>
  <script type="dojo/method" event="onchange">
    var txt = dojo.trim(this.tNode.value);
    console.log("CNG");
    if(txt.length == 0){
      this.tNode.value = this.defaultText;
    }
  </script>
</div>
```

จากตัวอย่างจะสังเกตได้ว่าในเอชทีเอ็มแอลแท็กจะปรากฏแอ็ททริบิวต์ (Attribute) และตำแหน่งของแท็กที่มีผลต่อการทำงานของจาวาสคริปต์ ซึ่งส่งผลให้จาวาสคริปต์มีผลในรูปแบบต่างๆกันไป แต่โดยลักษณะดังที่ปรากฏนี้อาจทำให้ผู้พัฒนาเกิดความสับสนจากลักษณะที่เพิ่มเติมจากเอ็ชทีเอ็มแอลแท็กทั่วไป จึงทำให้ไม่นำมาประยุกต์ใช้กับงานวิจัย



ลิขสิทธิ์มหาวิทยาลัยเชียงใหม่
Copyright© by Chiang Mai University
All rights reserved