

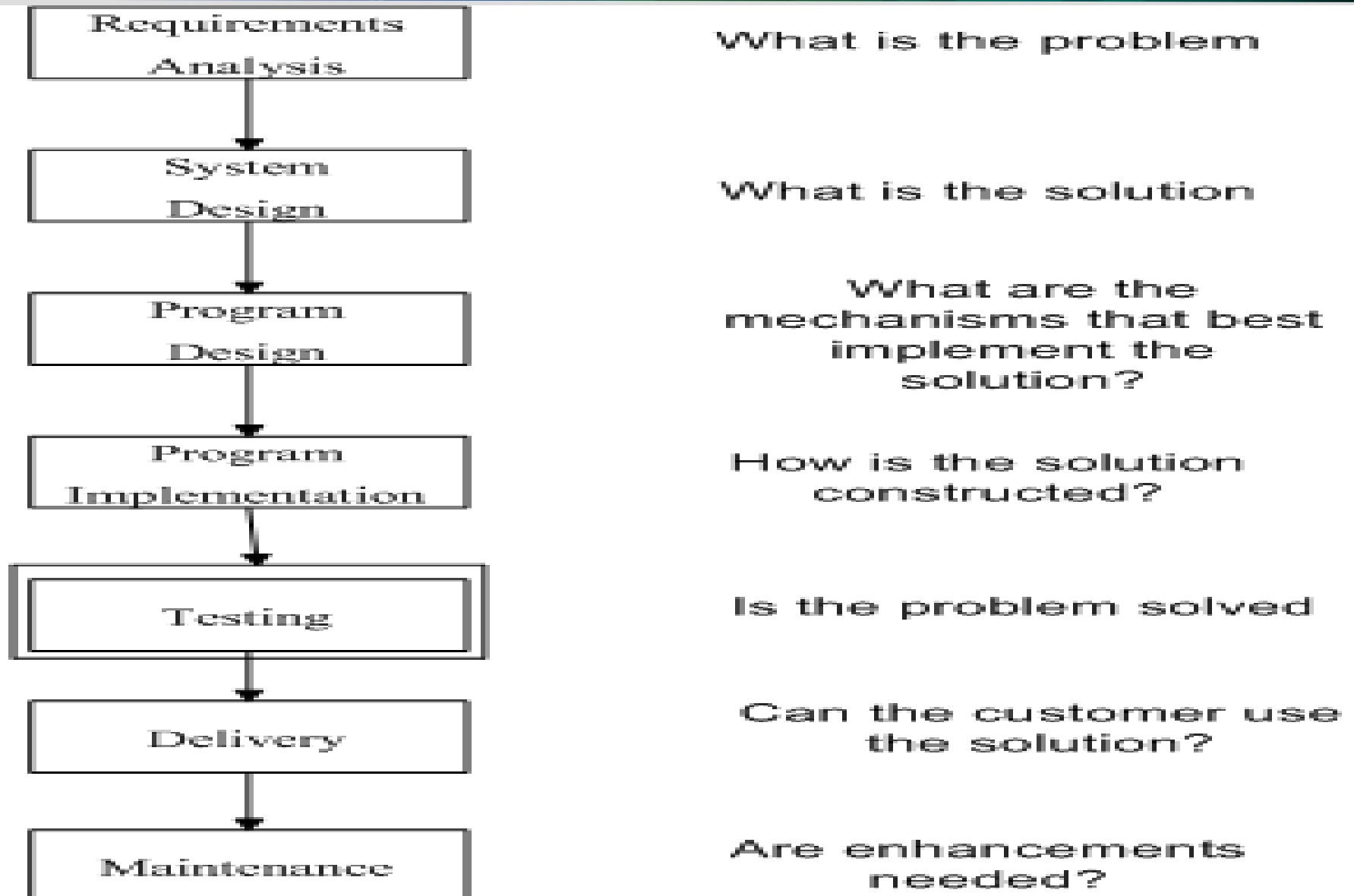
LOGO

ครั้งที่ 9

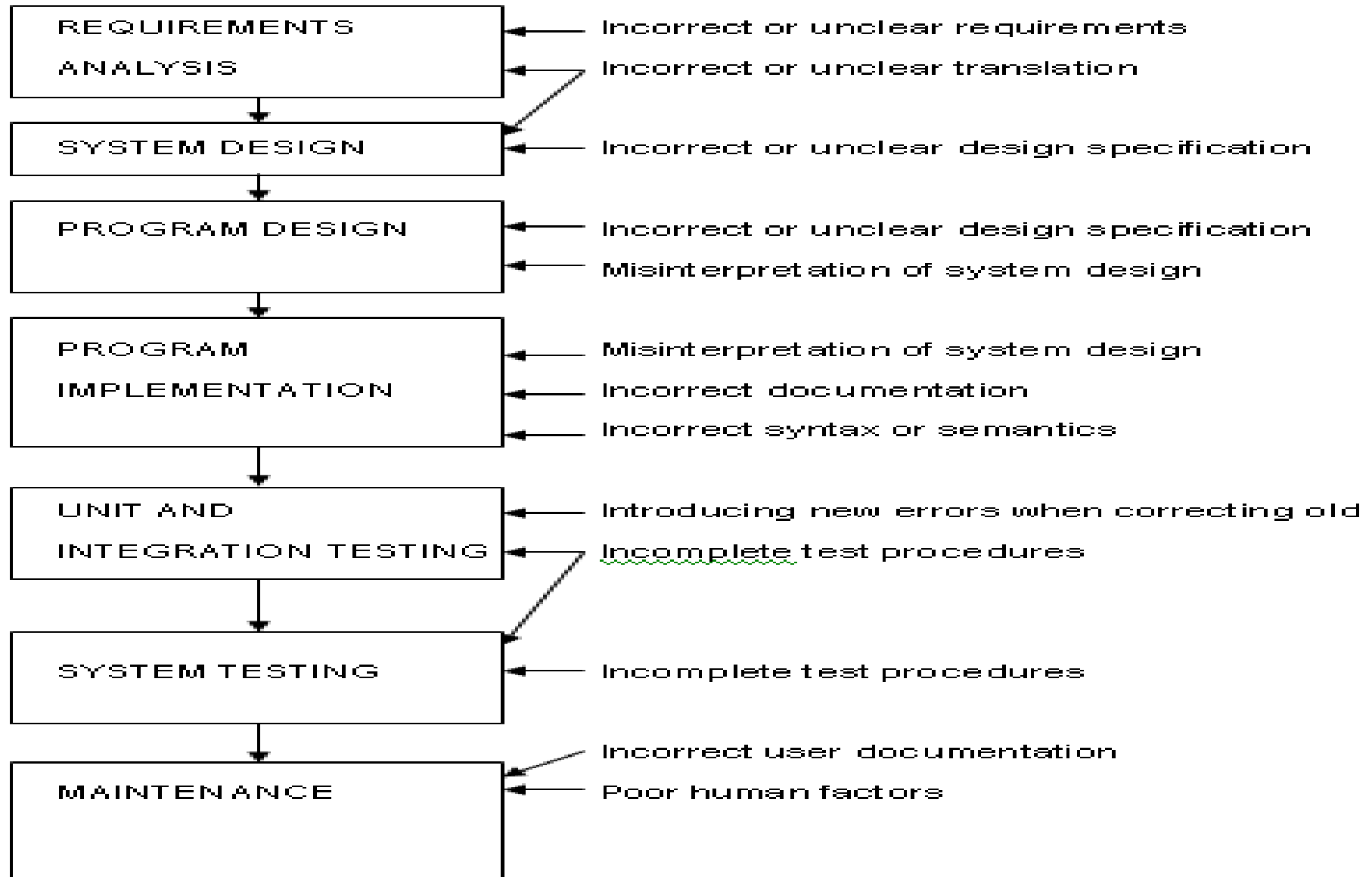
การทดสอบระบบ



The System Development Process



ข้อผิดพลาดที่เกิดขึ้นในการพัฒนาระบบ



ขั้นตอนในการทดสอบระบบ

1

Function testing

2

Performance testing

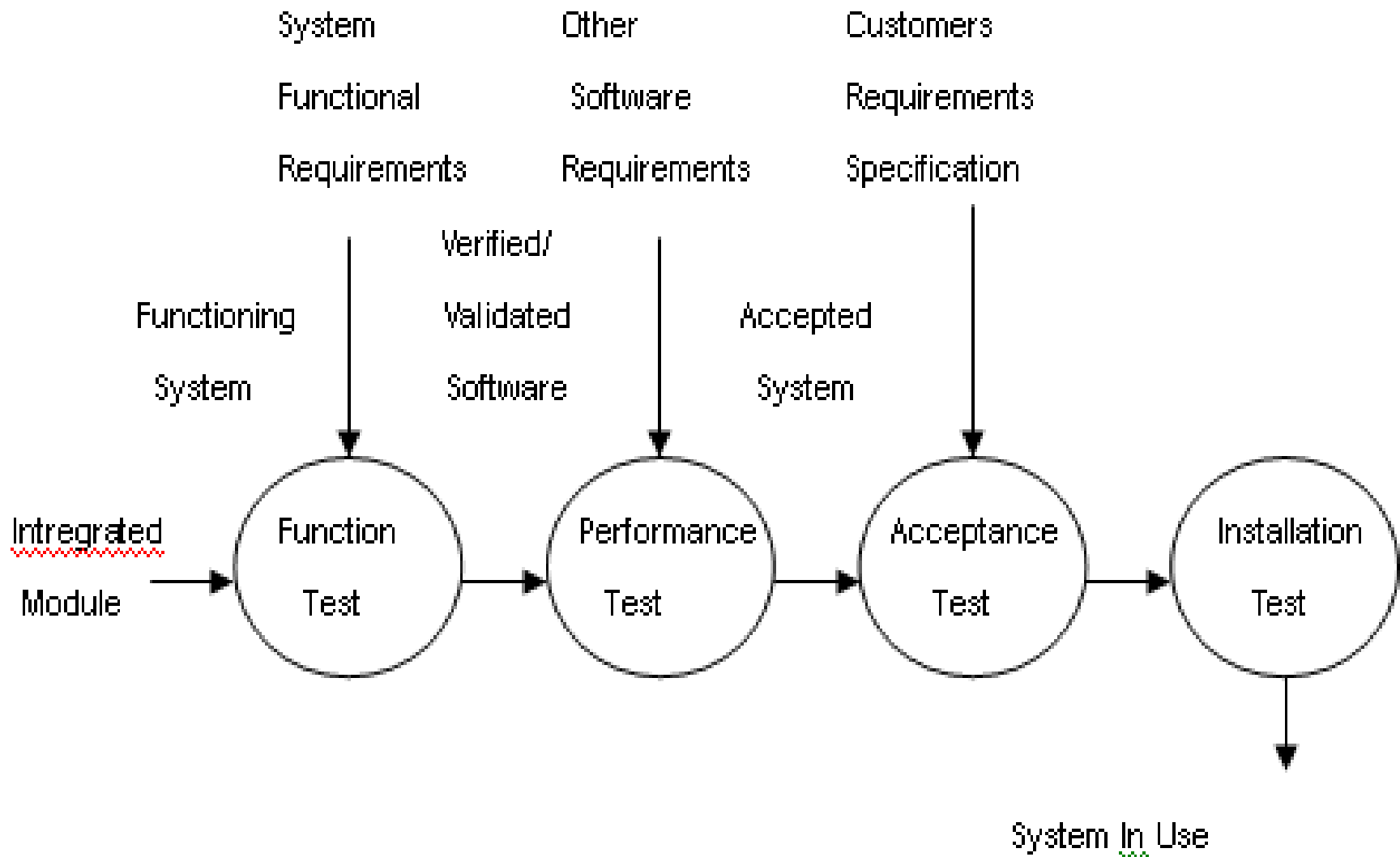
3

Acceptance testing

4

Installation testing

ขั้นตอนในการทดสอบระบบ



การทดสอบระบบเป็นระยะ (phased system testing)

❖ เป็นลำดับของระบบย่อย โดยมีการกำหนดขอบเขตหรือหน้าที่ที่ชัดเจน เป็น **functionality**

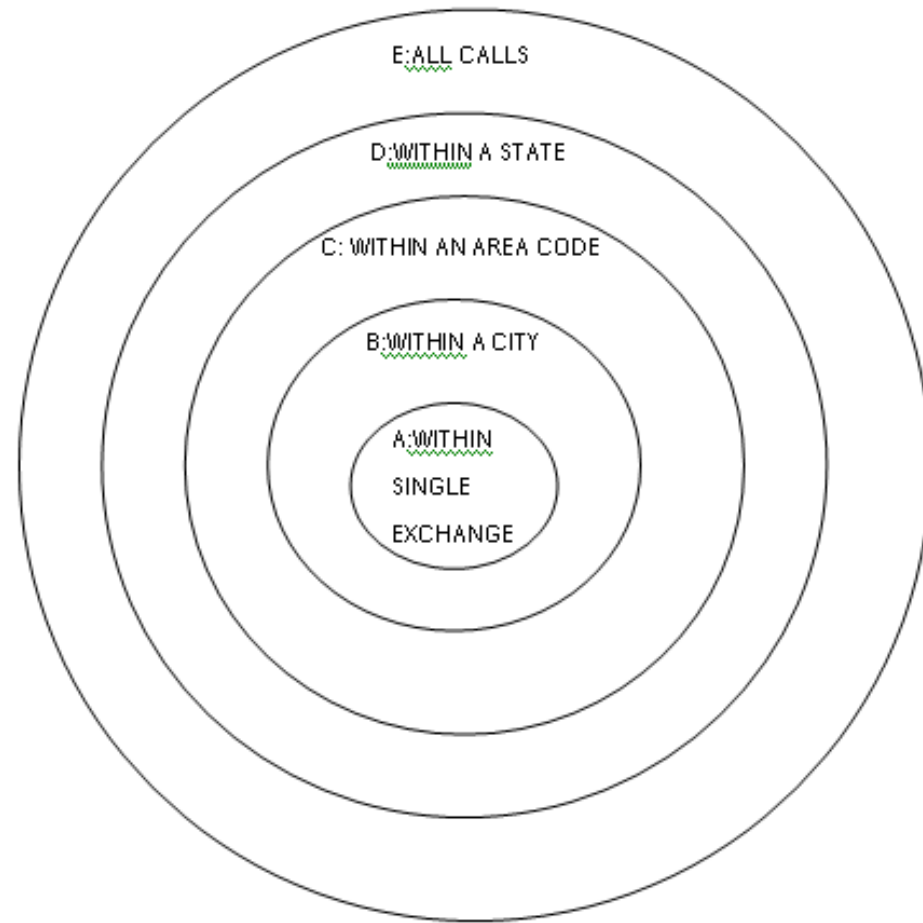
SystemA : การติดต่อภายในสาย

SystemB : การติดต่อภายในเมือง

SystemC : การติดต่อภายในพื้นที่

SystemD : การติดต่อภายในประเทศ

SystemE : การติดต่อทุกๆที่



ตัวอย่างระบบโทรศัพท์

ตารางที่ 8.1 Build Plan for Phone System

Spin	Function	Test Start	Test End
0	Calls within exchange	1 Sep 04	15 Sep 04
1	Calls within city	30 Sep 04	15 Oct 04
2	Calls within area code	25 Oct 04	5 Nov 04
3	Calls within State	10 Nov 04	20 Nov 04
4	All possible calls	1 Dec 04	15 Dec 04

Configuration Management Team

เป็นทีมงานที่ทำงานร่วมกับทีมงานทดสอบโดยทำหน้าที่
รับผิดชอบถึงการเปลี่ยนแปลงต่างๆที่เกิดขึ้นในระบบ

เมื่อมีข้อผิดพลาดเกิดขึ้น
ทราบถึงตำแหน่ง ผลกระทบ
ต่างๆที่เกิดขึ้นของการ
ดำเนินงานในระบบทั้งหมด



เป็นผู้รับผิดชอบต่อการ
เปลี่ยนแปลงระบบเพื่อให้
แน่ใจว่าการแก้ไขต่างะนั้น
ถูกต้อง ไม่ก่อให้เกิด
ข้อผิดพลาดใหม่ๆในระบบ
ตามมา

ซึ่งทีมงานนี้จะบันทึกถึงการดำเนินการทดสอบ
ผลของการทดสอบ
การเปลี่ยนแปลงต่างๆในรูปของเอกสาร (documentation)

FUNCTION TESTING

- ❖ ในขั้นตอนนี้เราจะไม่สนใจโครงสร้างของระบบแต่สนใจเฉพาะกิจกรรมที่ระบบสามารถกระทำได้นั้น
- ❖ ทดสอบการทำงานในแต่ละฟังก์ชันที่ตรงตามความต้องการของ **use case**

ตัวอย่าง

❖ การทดสอบระบบตรวจสอบระดับของน้ำในทะเลสาบ โดยระบบจะส่งข้อมูลให้แก่โอเพอร์เรเตอร์ เกี่ยวกับความปลอดภัยของระดับน้ำในทะเลสาบ โดยข้อมูลเข้าเป็นฟังก์ชันที่อยู่ในรูปแบบ ดังนี้ **INPUT :**
LEVEL(A,B) ซึ่ง **A** หมายถึงความสูงของน้ำหลังเขื่อนมีหน่วยเป็นฟุต
B หมายถึงจำนวนฝนตกใน24ชั่วโมงที่ผ่านมาหน่วยเป็นนิ้ว

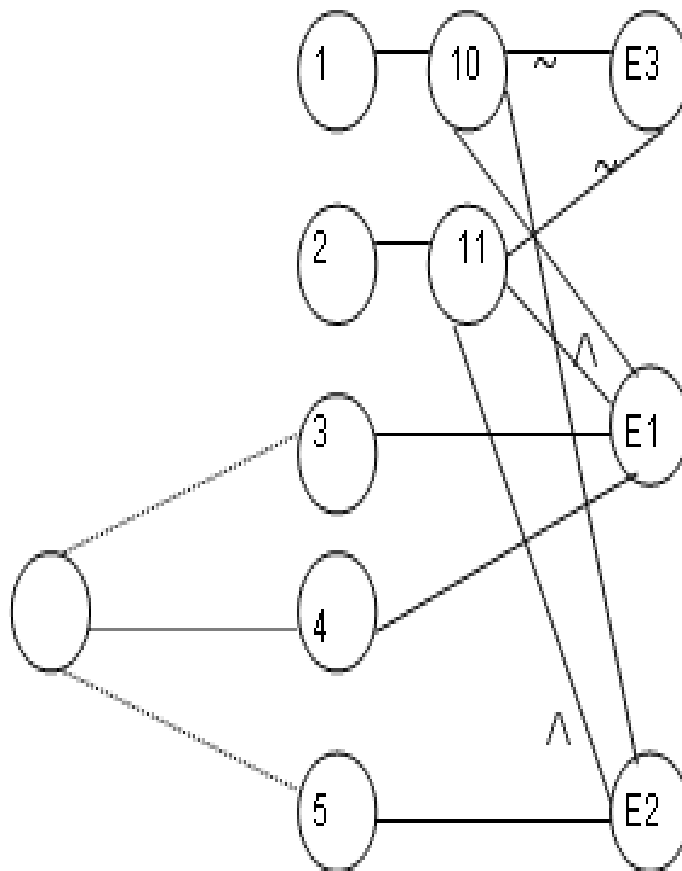
❖ การประมวลผล เป็นฟังก์ชันในการคำนวณถึงระดับของน้ำในเขื่อน ว่าอยู่ในช่วงที่ปลอดภัย หรือ ระดับน้ำสูงเกินไป หรือระดับน้ำอยู่ในระดับต่ำ

ตัวอย่าง

❖ ผลลัพธ์ จะแสดงเป็นสารสนเทศปรากฏบนหน้าจอ ดังนี้

- “LEVEL = SAFE” ถ้าระดับน้ำอยู่ในระดับปลอดภัยหรือต่ำ
- “LEVEL = HIGH” ถ้าผลลัพธ์น้ำอยู่ในระดับที่สูง
- “INVALID SYNTAX” ในกรณีอื่นๆ

Cause and Effect Graph ของ ฟังก์ชัน LEVEL



		Tests				
		1	2	3	4	5
Causes	1	I	I	I	I	S
	2	I	I	I	X	I
	3	I	S	S	X	X
	4	S	I	S	X	X
	5	S	S	I	X	X
Effects	E1	P	P	A	A	A
	E2	A	A	P	A	A
	E3	A	A	A	P	P

PERFORMANCE TESTING

- ❖ **Stress tests** เป็นการทดสอบความสามารถของระบบภายในช่วงเวลาที่กำหนด
- ❖ เช่น ทดสอบว่าระบบสามารถกระทำงานได้ภายใต้อุปกรณ์ทั้งหมดที่เชื่อมต่อ หรือ ผู้ใช้ที่ปฏิบัติงานทั้งหมด เป็นจำนวนสูงสุดเท่าใด ถ้าทดสอบแล้วได้ผลว่าสามารถทำงานพร้อมกันได้สูงสุด 30 คน แสดงว่าถ้ามีการใช้งานมากกว่านี้ระบบจะไม่มีประสิทธิภาพนั่นเอง

PERFORMANCE TESTING

- ❖ **Volume tests** เป็นการทดสอบจำนวนของข้อมูลมากที่สุดที่ระบบสามารถกระทำได้
- ❖ โดยตรวจสอบจากความต้องการเพื่อคำนวณถึงโครงสร้างข้อมูลต่างๆที่ระบบต้องปฏิบัติ รวมทั้งการปฏิบัติการต่างๆกับโครงสร้างข้อมูลต่างๆเหล่านั้น

PERFORMANCE TESTING

- ❖ **Configuration tests** เป็นการวิเคราะห์ถึง **Hardware** และ **Software** ที่สามารถปฏิบัติงานกับระบบ
- ❖ โดยตรวจสอบจากเอกสารระบุความต้องการ ทดสอบการปฏิบัติงานกับระบบคอมพิวเตอร์ต่างๆเหล่านั้นเพื่อให้แน่ใจว่าระบบสามารถกระทำได้จริงตามที่ได้ระบุไว้

PERFORMANCE TESTING

- ❖ **Compatibility tests** เป็นการทดสอบที่จำเป็นสำหรับระบบที่มีการปฏิสัมพันธ์กับระบบอื่นๆ
- ❖ โดยเฉพาะอย่างยิ่งการเข้าถึงข้อมูล การดึงข้อมูล หรือการอ่านข้อมูล การทดสอบนี้จะทดสอบในแง่ของความเร็ว ความถูกต้องในการดึงข้อมูลจากระบบฐานข้อมูลนั่นเอง
- ❖ นอกจากนี้การทดสอบนี้มีผลต่อการค้าเพราะทดสอบว่าซอฟต์แวร์นี้ใช้ร่วมกับซอฟต์แวร์อื่นในท้องตลาดได้หรือไม่

PERFORMANCE TESTING

- ❖ **Regression tests** การทดสอบนี้เป็นสิ่งจำเป็นอย่างยิ่งสำหรับการนำระบบงานใหม่แทนที่ระบบงานที่กำลังปฏิบัติงานอยู่ เพื่อรับประกันว่าระบบใหม่มีประสิทธิภาพดีกว่า ซึ่งใช้ทดสอบสำหรับการพัฒนาที่มีหลายระยะ (phased development) หรือทดสอบโปรแกรมส่วนที่ได้รับการแก้ไขเฉพาะส่วน
- ❖ การทดสอบความถดถอย, **regression test** จะเป็นตัวหาว่าเมื่อแก้ไขโค้ดเพื่อเพิ่มความสามารถใหม่+แก้บั๊กแล้ว ความสามารถเดิมเสียไป หรือเพิ่มบั๊กใหม่ หรือไม่

PERFORMANCE TESTING

- ❖ **Security tests** เป็นการทดสอบความปลอดภัยของระบบ ซึ่งได้กำหนดไว้ในเอกสารความต้องการ
- ❖ โดยตรวจสอบการเข้าถึงในแง่ของฟังก์ชันการทำงาน
- ❖ การเข้าถึงข้อมูลต่างๆ ของผู้ใช้ระดับต่างๆ

PERFORMANCE TESTING

❖ **Timing tests** คำนวณถึงเวลาตอบสนองกับผู้ใช้ที่กระทำ
หน้าที่ต่างๆของระบบ

PERFORMANCE TESTING

- ❖ **Environmental tests** เป็นการพิจารณาความสามารถของระบบว่าสามารถทำงาน ณ สถานที่ที่ตั้งได้หรือไม่ ซึ่งการติดตั้งอาจมีปัญหามาจากความร้อน ความชื้น สารเคมี สัญญาณไฟฟ้า หรือสภาพแวดล้อมที่ส่งผลกระทบต่อการทำงานของระบบ

PERFORMANCE TESTING

- ❖ **Quality tests** เป็นการคำนวณคุณภาพของซอฟต์แวร์ ในเรื่องของความน่าเชื่อถือ การบำรุงรักษาระบบ และการได้มาของระบบ

PERFORMANCE TESTING

- ❖ **Recovery tests** เป็นการทดสอบการคืนคืนข้อมูล หรือ ทดสอบการตอบสนองของระบบในกรณีเกิดความผิดพลาดของข้อมูล หรือ อุปกรณ์หรือ กำลังไฟ เป็นต้น

PERFORMANCE TESTING

- ❖ **Maintenance tests** เป็นการทดสอบการบำรุงรักษาระบบ ในกรณีต้องการเครื่องมือหรือกระบวนการที่ช่วยสำหรับวิเคราะห์ความผิดพลาดของระบบ
- ❖ โดยตรวจสอบถึงเครื่องมือต่างๆที่ได้ระบุไว้ เช่น โปรแกรมวิเคราะห์ความผิดพลาด แผนที่หน่วยความจำ การติดตามรายการปฏิบัติงานไคอะแกรมของวงจร
- ❖ เครื่องมืออื่นๆ เพื่อตรวจสอบว่าเครื่องมือต่างๆเหล่านี้มีอยู่จริง และสามารถนำมาช่วยในการบำรุงรักษาระบบได้

PERFORMANCE TESTING

- ❖ **Documentation tests** ตรวจสอบเอกสารต่างๆที่จำเป็น และได้ระบุไว้ในเอกสารระบุความต้องการ
- ❖ พิจารณาและตรวจสอบว่าระบุได้อย่างถูกต้อง และคงที่ ง่ายต่อการอ่านหรือไม่

PERFORMANCE TESTING

- ❖ **Human factor tests** เป็นการทดสอบการแสดงผลทางจอภาพ
- ❖ ข่าวสารต่างๆที่แสดง รูปแบบของรายงาน ว่าชัดเจนหรือสวยงามหรือไม่
- ❖ ง่ายต่อผู้ใช้ หรือไม่

ACCEPTANCE TESTING

❖ เป็นการนำระบบไปให้ลูกค้าหรือผู้ใช้ทดสอบ โดยให้ลูกค้ากำหนดถึงกรณีทดสอบต่างๆได้ตามความต้องการ

ACCEPTANCE TESTING

Benchmark test

การวัดเปรียบเทียบสมรรถนะ
เกณฑ์มาตรฐาน
เตรียมกลุ่มของกรณีทดสอบซึ่ง
แทนการปฏิบัติงานของระบบ
ลูกค้าจะทดสอบประสิทธิภาพ
ของระบบในแต่ละกรณีทดสอบ
โดยใช้ทีมงานทดสอบที่
ปฏิบัติงานเฉพาะกรณีก็ได้
วิธีการนี้ลูกค้าสามารถใช้สำหรับ
ความต้องการพิเศษก็ได้

Pilot test

ผู้ใช้งานจะทำการทดสอบ
ระบบในสิ่งที่จะต้องกระทำซ้ำๆกัน
ทุกวัน(**everyday
working**) ทดสอบทุกๆ
หน้าที่ หรือนำข้อมูลตัวอย่าง
จำนวนหนึ่งมาประมวลผล เพื่อ
จะได้ประเมินผล แล้วนำไป
ปรับปรุงกระบวนการที่จะ
ทดสอบจริงต่อไป

Parallel test

เป็นการทดสอบโดยระบบเก่า
และระบบใหม่ทำงานขนานกัน
ไป ผู้ใช้ระบบสามารถ
เปรียบเทียบ และทดสอบการ
ทำงานของระบบใหม่กับระบบ
เก่า เพื่อให้แน่ใจว่าระบบเก่า
สามารถทำงานมีประสิทธิภาพ
และแทนที่ระบบเก่าได้

Alpha Testing

- ❖ ใช้ข้อมูลสมมติในการทดสอบ สมมติสถานการณ์ที่มีการทดสอบ 4 ประการ คือ
 - การทดสอบการกู้คืนข้อมูล (Recovery Testing) : ทดสอบกรณีระบบล้ม
 - การทดสอบความปลอดภัย (Security Testing) : ทดสอบกรณีมีการลักลอบการเรียกใช้ข้อมูล
 - การทดสอบความกดดัน (Stress Testing) : ทดสอบประสิทธิภาพของระบบกรณีอยู่ภายใต้ความกดดัน เช่น การเข้าใช้งานพร้อมกันของผู้ใช้หลายคน
 - การทดสอบประสิทธิภาพการทำงาน (Performance Testing) : ทดสอบประสิทธิภาพการทำงานภายใต้สภาพแวดล้อมที่แตกต่างกัน เช่น ระบบปฏิบัติการ

Beta Testing

- ❖ ใช้ข้อมูลจริงในการทดสอบ ภายใต้สถานการณ์จริง
- เป็นการซ้อมการติดตั้งระบบเพื่อใช้งานจริง
- เป็นการทดสอบระบบอย่างสมจริง ทั้งสถานการณ์ ข้อมูล ขั้นตอนการทำงาน เอกสารคู่มือ การฝึกอบรม การสนับสนุนการทำงาน รวมทั้งการแก้ไขปัญหาจากการทดสอบแบบ **Alpha** ด้วย

INSTALLATION TESTING

- ❖ การทดสอบการติดตั้งระบบ แต่ถ้าในขั้นตอนที่แล้วทดสอบในสถานที่ตั้งอยู่แล้วขั้นตอนนี้ไม่จำเป็น
- ❖ การทดสอบในขั้นตอนนี้เป็น การทดสอบถึงอุปกรณ์ที่นำมาใช้ร่วมกับระบบ นั้นสามารถติดต่อหรือทำงานร่วมกับระบบได้หรือไม่ หน้าที่ต่างๆที่ระบบกระทำได้รวมทั้งการปฏิบัติการกับข้อมูลการเข้าถึงข้อมูลต่างๆถูกต้องหรือไม่
- ❖ ผลของการทดสอบคือ ระบบสามารถกระทำได้โดยสมบูรณ์ หรือ ต้องแก้ไข ซึ่งอาจมีผลมาจากเงื่อนไขของสถานที่ติดตั้ง

การติดตั้งระบบ (Implement)

- ❖ ติดตั้งเพื่อใช้งานใหม่ทันที (**Direct Changeover**)
- ❖ การติดตั้งระบบแบบคู่ขนาน (**Parallel Running**)
- ❖ การติดตั้งระบบแบบโครงการนำร่อง (**Single Location Changeover/Pilot Project**)
- ❖ การติดตั้งระบบแบบเป็นช่วงๆ (**Phased Changeover**)

ติดตั้งเพื่อใช้งานใหม่ทันที (Direct Changeover)

- ❖ ยกเลิกการใช้งานระบบเก่าและเริ่มใช้ระบบงานใหม่ทันที
- ❖ เสียค่าใช้จ่ายน้อย มีความเสี่ยงสูง
- ❖ สามารถเลื่อนกำหนดการใช้งานระบบใหม่ออกไปได้หากองค์กรหรือผู้ใช้งานยังไม่มีความพร้อมสำหรับระบบใหม่

การติดตั้งระบบแบบคู่ขนาน (Parallel Running)

- ❖ ใช้งานระบบใหม่ไปพร้อมกับระบบเก่า
- ❖ สามารถเปรียบเทียบผลของการดำเนินงานระหว่างระบบใหม่กับระบบเก่าได้
- ❖ เสียค่าใช้จ่ายสูง
- ❖ ผู้ใช้อาจเกิดความสับสน

การติดตั้งระบบแบบโครงการนำร่อง

(Single Location Changeover/Pilot Project)

- ❖ ใช้ระบบงานใหม่เพียงหน่วยเดียวก่อน แล้วจึงค่อยปรับเปลี่ยนทั้งหมดเมื่อทุกอย่างลงตัว
- ❖ เสียค่าใช้จ่ายน้อยกว่า **2** แบบแรก
- ❖ ความเสียหายจะถูกจำกัดอยู่ที่หน่วยๆ เดียว
- ❖ สามารถติดตามผลและดูแลระบบใหม่ได้อย่างเต็มที่

การติดตั้งระบบแบบเป็นช่วงๆ

(Phased Changeover)

- ❖ ใช้ระบบงานใหม่เพียงบางส่วนก่อนระยะหนึ่งควบคู่ไปกับระบบงานเก่า แล้วจึงค่อยทยอยใช้ระบบงานใหม่เพิ่มขึ้นทีละส่วนจนกระทั่งครบทุกส่วนอย่างเต็มรูปแบบ
- ❖ จำกัดความเสี่ยงได้
- ❖ ไม่สอดคล้องกับการดำเนินงานในระบบใหม่และระบบเก่า

TEST TOOLS

Simulator

1

เป็นเครื่องมือที่ช่วยแสดง
คุณลักษณะทั้งหมดของ
อุปกรณ์และระบบ แต่เป็น
การจำลองไม่มีเครื่องมือหรือ
อุปกรณ์จริงๆ
เช่น การจำลองการบิน

Monitor

2

เป็นอุปกรณ์หรือซอฟต์แวร์ใน
การดักจับข้อมูลที่ผ่านมาจาก
โปรเซสหนึ่งไปยังอีกโปรเซส
หนึ่ง ช่วยให้เราค้นหาหรือ
ติดตามแหล่งของความ
ผิดพลาดต่างๆที่เกิดขึ้นเพื่อ
แก้ไขได้ง่ายขึ้น

Analyzer

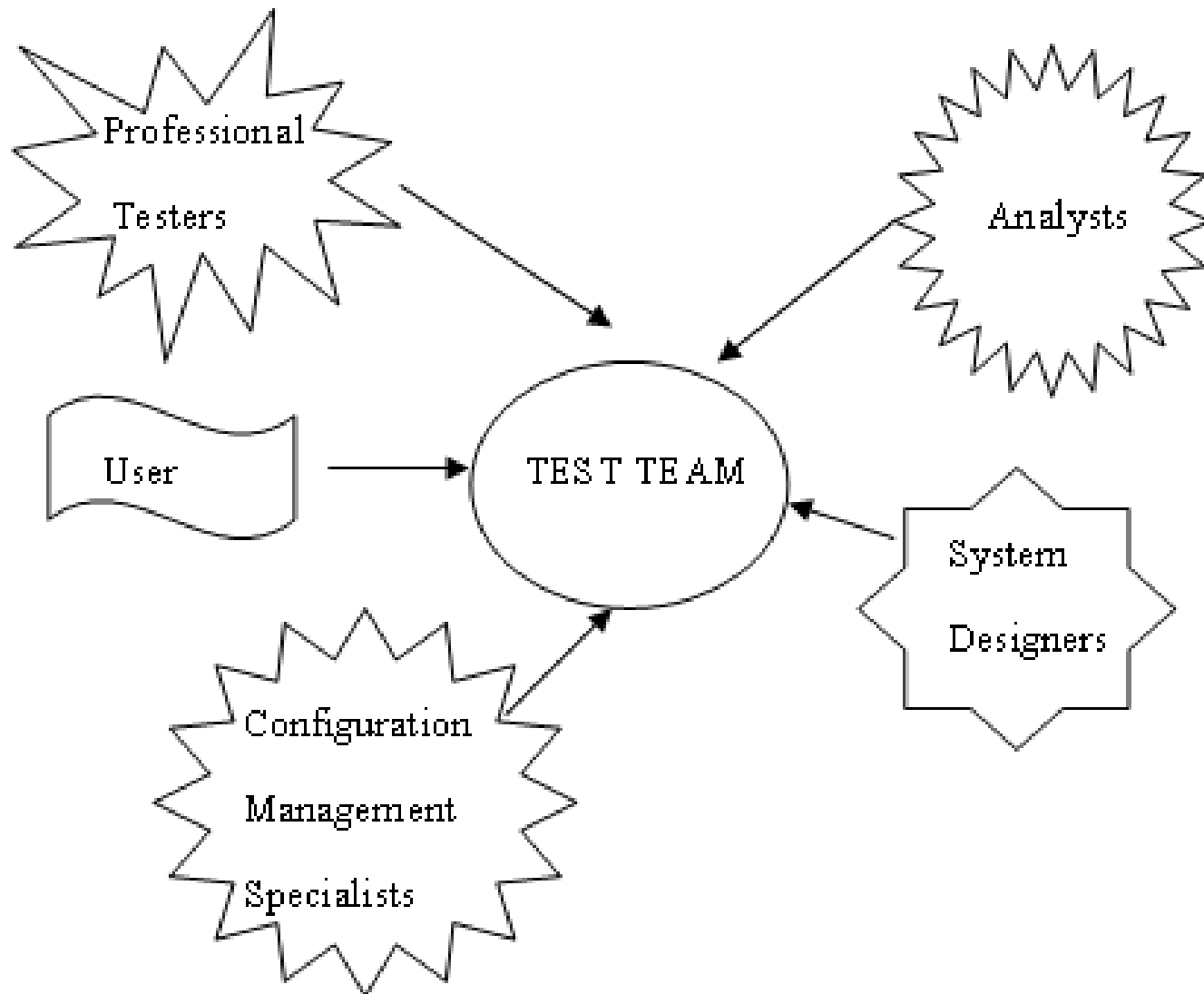
3

เป็นเครื่องมือในการวิเคราะห์
ข้อมูลต่างๆตามระเบียบที่
กำหนด เช่นวิเคราะห์จำนวน
คำสั่งต่างๆที่ปฏิบัติงานใน
ระหว่างการทดสอบ รวมทั้งรู้
ทันและคำสั่งที่ไม่ได้
ปฏิบัติงาน

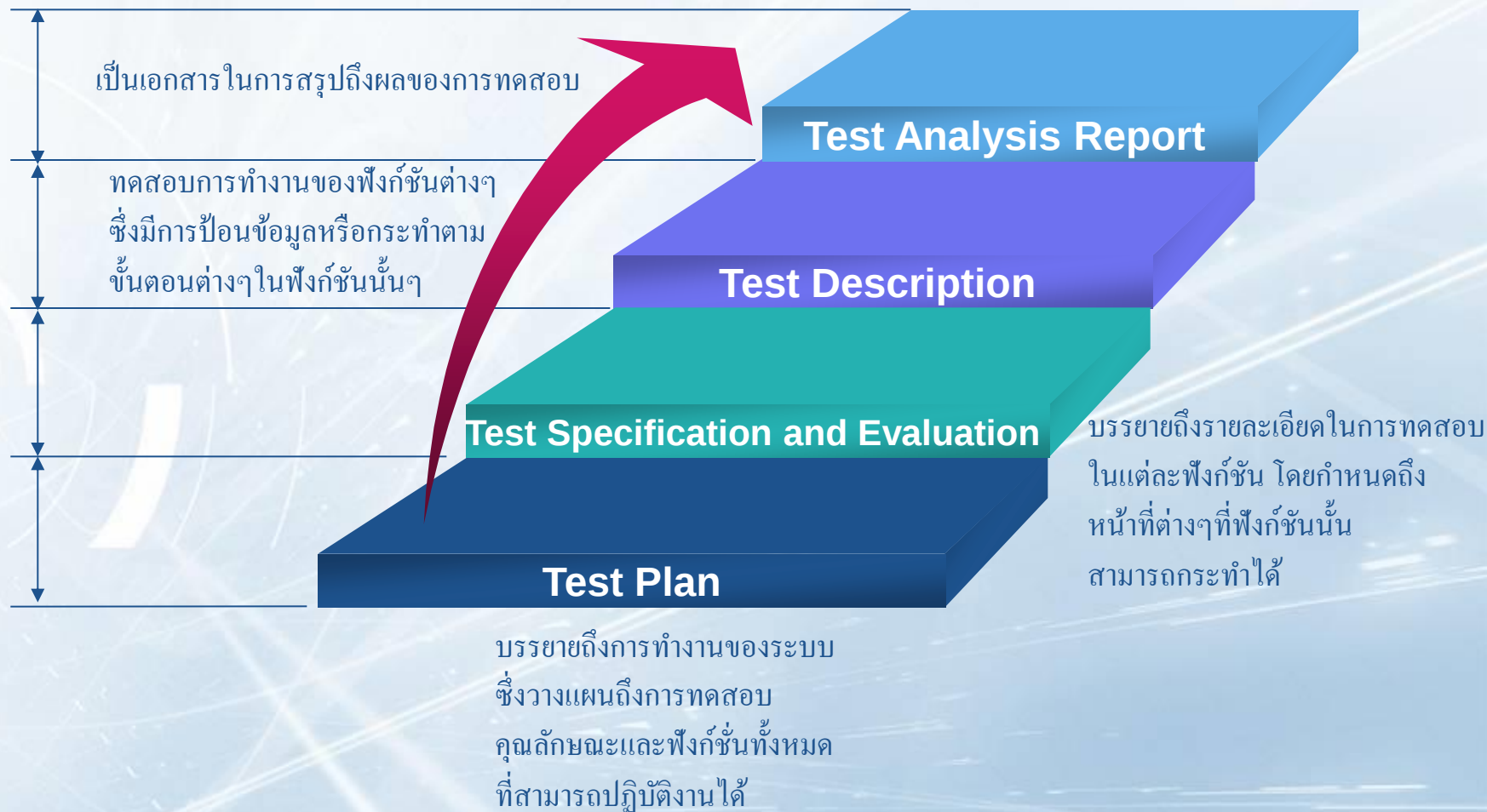
TEST TEAM



TEST TEAM



TEST DOCUMENTATION

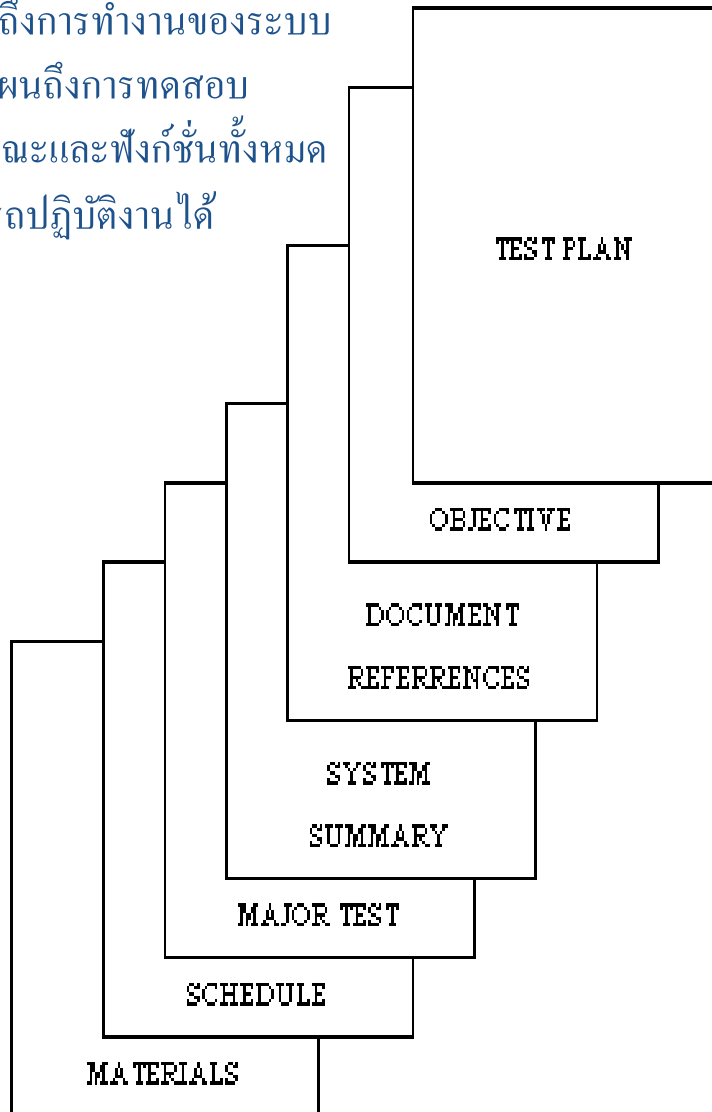


ความสัมพันธ์ของเอกสารต่างๆในกระบวนการทดสอบ



Test Plan Components

บรรยายถึงการทำงานของระบบ
ซึ่งวางแผนถึงการทดสอบ
คุณลักษณะและฟังก์ชันทั้งหมด
ที่สามารถปฏิบัติงานได้



กำหนดวัตถุประสงค์ (Objective) โดยรายละเอียดประกอบไปด้วย

- แนะนำวิธีการจัดการทดสอบ
- แนะนำเทคนิคต่างๆที่ใช้ในระหว่างการทดสอบ
- สร้างแผนงานและกำหนดระยะเวลาในการทดสอบ รวมทั้งรายละเอียดของอุปกรณ์ที่จำเป็น วิธีการทดสอบ ผลลัพธ์ที่ต้องการ
- อธิบายลักษณะและขอบเขตของแต่ละการทดสอบ
- อธิบายถึงหนทางของการทดสอบที่สามารถกระทำการได้สำเร็จและสามารถประเมินค่าของหน้าที่และประสิทธิภาพของระบบได้
- ข้อมูลในการนำเข้า ของแต่ละฟังก์ชันทดสอบ ผลที่คาดว่าจะได้รับ

TEST SPECIFICATION AND EVALUATION

- ❖ การทดสอบระบบนี้จะมีการแบ่งระบบออกเป็นระบบย่อยๆ และในแต่ละระบบย่อย ต้องมีการสร้างรายละเอียดของการทดสอบและการคำนวณการทำงาน (test specification and evaluation) เริ่มจากเขียนความต้องการที่ต้องการจากระบบย่อยนี้ ต่อจากนั้นพิจารณาถึงความต้องการต่างๆที่ได้กำหนดขึ้นนำมาสร้างวัตถุประสงค์ของการทดสอบ

TEST SPECIFICATION AND EVALUATION

❖ การเขียนความต้องการต่างๆจะแสดงในส่วนหัวตาราง ส่วนหน้าที่ต่างๆที่ทำการทดสอบจะเขียนทางซ้ายของแต่ละแถว

❖ สำหรับ X แทนถึงกิจกรรมต่างๆที่ระบบสามารถกระทำเพื่อให้ได้ตามวัตถุประสงค์ที่กำหนด โดยต้องระบุถึงเงื่อนไขของการทดสอบให้ชัดเจน

- การรับเข้าข้อมูลเข้าสู่ระบบว่ามาจากผู้ใช้หรืออุปกรณ์ใดหรือจากสร้างจากโปรแกรมหรืออุปกรณ์ใด
- การทดสอบนั้นต้องให้ครบทุกๆหน้าที่ทุกๆทางเลือก ทุกๆส่วน
- ข้อมูลมีการบันทึกอย่างไร
- ถ้าการทดสอบมีชุดหรือลำดับของการทดสอบย่อยๆ ต้องทราบถึงลำดับในการทดสอบ

Test	Requirement		
	Generate And Maintain Data Base	Selectively Retrieve Data	Produce Specialized Reports
1.Add new record	X		
2.Add field	X		
3.Change field	X		
4.Delete record	X		
5.Delete field	X		
6.Create index		X	
Retrieve record with a request			
7.Cell number		X	
8.Water height		X	
9.Canopy height		X	
10.Ground cover		X	
11.Percolation rate		X	
12.Print full data base			X
13.Print Directory			X
14.Print Keywords			X
15.Print simulation			

Test Description

❖ เป็นเอกสารที่แสดงถึงการทดสอบข้อมูลในส่วนรายละเอียด หลังจาก
ระบุถึงการทดสอบในฟังก์ชันย่อยต่างๆแล้ว เอกสารนี้เป็นข้อเสนอแนะ
ในการทดสอบ ซึ่งเริ่มจากการเปิดเครื่องการ โต้ตอบกับระบบ ใน
ขั้นตอนต่างๆตามลำดับ ซึ่งต้องมีรายละเอียดที่ละเอียดและชัดเจน อัน
ประกอบด้วย

- Mean of Control
- Data
- Procedure

Test Script for Change Field Function

Step N :	Press function key 4 : Access data file
Step N+1 :	Screen will ask for name of data file. Type 'sys.test.txt'
Step N+2 :	Menu will appear, reading: - Delete file - Modify file - Rename file Place cursor next to 'Modify file ' and press RETURN key.
Step N+3 :	Screen will ask for record number. Type '4016'
Step N+4 :	Screen will fill with data fields for record 4017 : RECORD NUMBER : 4017 CELL X : 0042 CELL Y : 0036 SOIL TYPE : CLAY PERCOLATION RATE : 4 FT/HR VEGETATION : KUDZU CANOPY HEIGHT : 25 FT WATER TABLE : 12 FT CONSTRUCT : OUTHOUSE MAINTENANCE CODE : 3T/4F/9R
Step N+5 :	Press function key 9 : Modify.
Step N+6 :	Entries on screen will be highlighted. Move cursor to VEGETATION

จากตัวอย่าง ลำดับของการทดสอบรายละเอียดนั้นถูกกำหนดตามลำดับของตัวเลขที่ระบุ โดยอธิบายถึงเหตุการณ์ต่างๆที่เกิดขึ้นจริง อาทิเช่นการกดคีย์ต่างๆ การแสดงผลทางจอภาพ ผลของการทำงานที่ได้ อุปกรณ์ต่างๆที่เกี่ยวข้อง รายงานที่ได้รับ เป็นต้น

Test Analysis Report

❖ เป็นเอกสารที่บรรยายผลลัพธ์ของการทดสอบ ซึ่งเป็นเอกสารใช้สำหรับ
วิเคราะห์การทดสอบ ผลของการทดสอบอาจถูกต้อง หรือเกิด
ข้อผิดพลาด

รายงานที่แสดงถึงความผิดพลาดที่เกิดขึ้นเรียกว่า **discrepancy report form**

- ❖ **state** กล่าวถึงระบบก่อนที่เกิดข้อผิดพลาดขึ้นในระบบ
- ❖ **evidence** เหตุการณ์ที่เกิดข้อผิดพลาดขึ้น
- ❖ **action** กิจกรรมหรือการกระทำที่เกิดขึ้นปรากฏขึ้นเมื่อมีความผิดพลาดเกิดขึ้นในระบบ
- ❖ **should** อธิบายถึงรายละเอียดที่ระบบควรจะทำโดยปราศจากความผิดพลาด
- ❖ **requirements** มีการอ้างอิงที่เกี่ยวข้องกับความต้องการที่ระบุหรือกำหนดไว้
- ❖ **impact** ผลกระทบของการเกิดความผิดพลาดนี้ที่เกิดขึ้นในระบบ
- ❖ **severity** ระดับของการแก้ปัญหาหรือความรุนแรงที่เกิดขึ้น ถ้าสามารถกระทำได้

discrepancy report form (DRF)

รูปภาพที่ 8.12 แสดงถึง Discrepancy Report Form

DRF Number :		Tester Name :	
Date :		Time :	
Text Number :			
Script step executed when error occurred :			
Description			
.....			
.....			
Activities before occurrence of error			
.....			
.....			
Expected results :			
.....			
Requirements affected :			
.....			
Impact of error on test :			
.....			
Impact of error on system :			
.....			
Severity Level : Low 1 2 3 4 5			