

IFPUG 1997 Fall Conference, Scottsdale, September 15-19

Adapting Function Points to Real-Time Software

D. St-Pierre

A. Abran

M. Araki

J.M. Desharnais

Agenda

- A - Project history (*Desharnais*)
- B - Quality characteristics of measurement methods (*Abran*)
- C - Lessons from previous attempts (*Abran*)
- D - Full Function Points (FFP): FPA extension for Real-Time Software (*St-Pierre*)
- E - Industry field tests in Japan (*Araki*)
- F - Conclusion (*Abran*)

A - Project History

A1 - Project Objectives

A2 - Project Structure

A3 - Project Roles

A4 - Project Steps

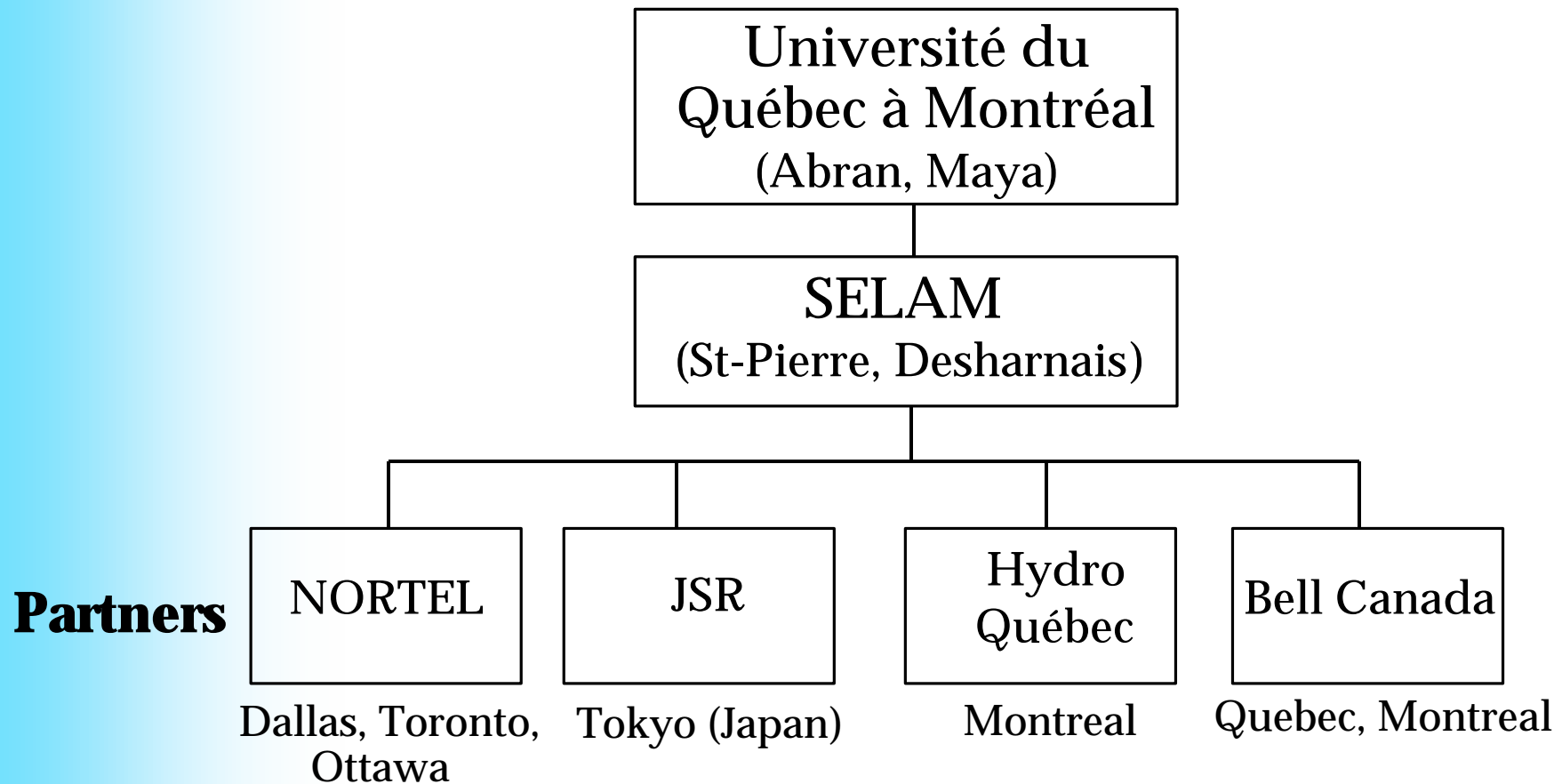
A5 - Project Public Deliverables

A6 - Project Services

A1 - Project Objectives

- Measure **adequately** the functional size of real-time software
- Build on IFPUG assets
- Facilitate migration path and ease of transition for organization historical database
- Keep IFPUG relevant
- Be aligned with ISO framework

A2 - Project Structure



All partners have many thousands employees and billions of dollars of revenue each year

A3 - Project Roles

- UQAM: Research Management
- SELAM: Function Points Expertise and Applied Research
- Industry Partners:
 - ❖ Data collection sites
 - ❖ Expertise in real-time
 - ❖ Funding

A4 - Project Steps (1995-1997)

- Identification of proposed techniques
- Identification of industrial partners
- Re-design of counting structure
- Field tests
- Partner sites reports
- Initial public release (WEB site, March 97)
- Full public release (IFPUG, Fall 97)

A5 - Project Public Deliverables

*“Full Function Points:
Counting Practices Manual”*

- Counting Structure
- Counting Rules
- Counting Examples

A6 - Project Services

- A Full Case Study
- FFP Training Services
- FFP Counting Support

B - Quality Characteristics of Measurement Methods

- Key Concepts of Measurement Methods

Ref: «*From Software Metrics to Software Measurement Methods: A Process Model*»

Jacquet & Abran, Third International Symposium and Forum on Software Engineering Standards, ISESS'97, Walnut Creek (CA). June 2-6, 1997

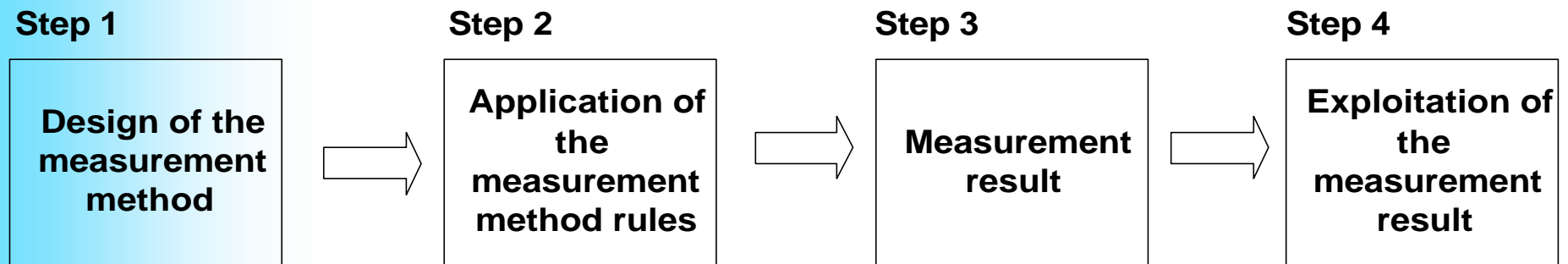
- ISO Quality requirements

Ref: *Information technology - Software measurement - Verification of a Functional Size Measurement Method*

Project Editor: A. Abran

ISO/IEC JTC1/SC7 WG12 Sub-project 7.31.3, June 1997

B1 - Key Concepts: Measurement Method Model



- 1) Before measuring, you need a measurement method
- 2) The rules of the measurement method are applied to software (or piece of software)
- 3) Output of Step 2 is a result (it should be auditable)
- 4) The result is exploited (quantitative or qualitative)

B1 - Key Concepts: Metrics ?? = ?? Measurement Methods

Too many *metrics* are discussed
from a Step 4 perspective only
without prior investigation of
the foundations of measurement methods!!

B1 - Key Concepts: The right Sequence!

Everybody is interested in Step 4 but:

- Step 3 must have been accurate

and

- Step 2 must have been carried out correctly

and

- Step 1 must be **VALID!**

(and in that sequence)

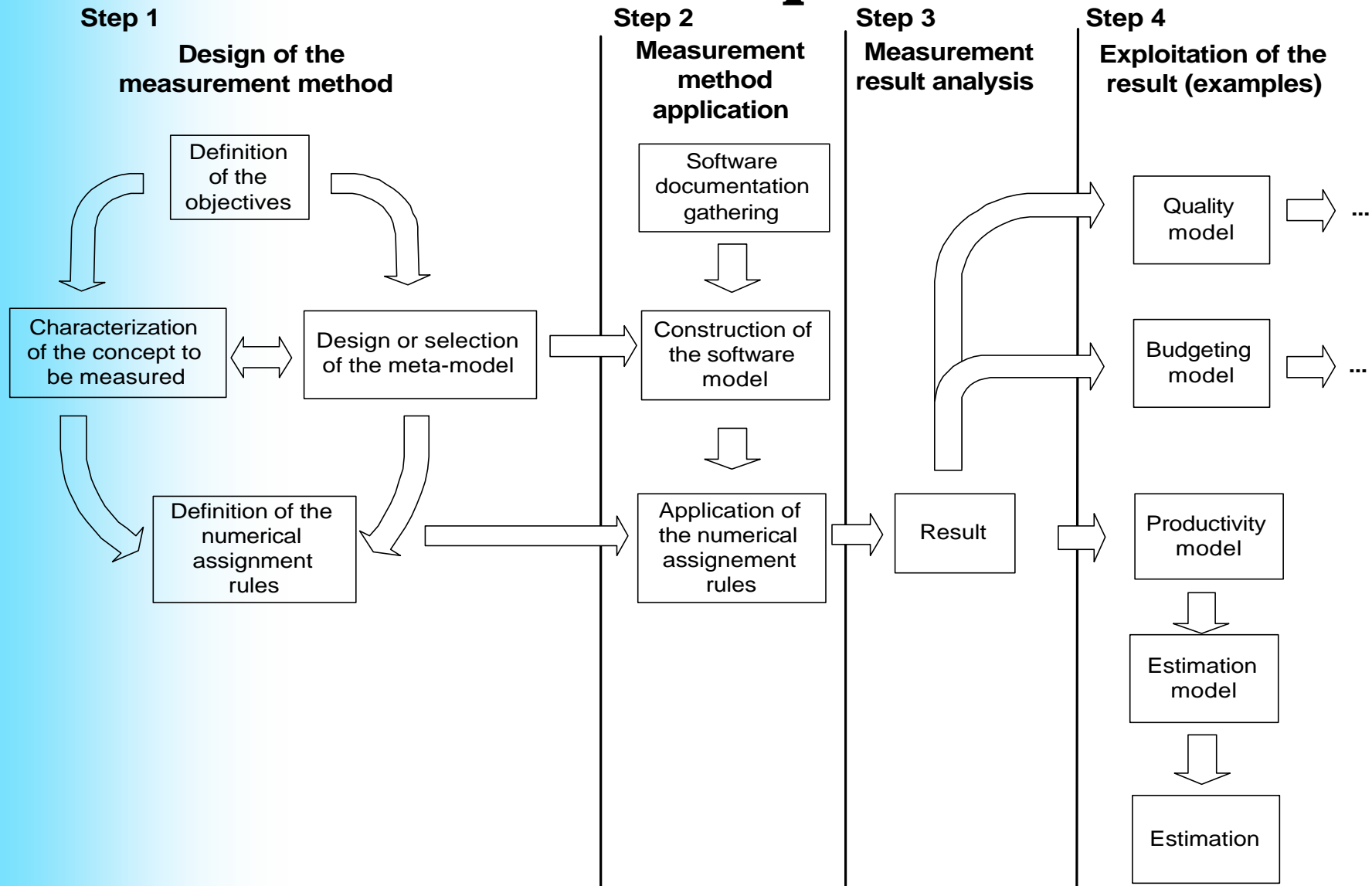
B1 - Key Concepts: Research Strategy

Project strategy accepted by research partners:

- Phase A (included):
 - ❖ Step 1 (Measurement Design)
 - ❖ Step 2 (Measurement in practice)

- Phase B (later):
 - ❖ Step 3 (Results analysis)
 - ❖ Step 4 (Productivity, Estimation & Quality Models)

B1 - Key Concepts: Measurement Sub-steps



B2 - ISO Quality Requirements

- Measurement Design:
 - ❖ Consistency with concept: Functional Size
 - ❖ Not ambiguous & relationships clearly defined
- Measurement Practice:
 - ❖ Documentation required
 - ❖ Identification of valid components, etc.
- Measurement Results:
 - ❖ Reliability
 - ❖ Repetitiveness, etc.
- Convertibility

C - Lessons from Previous Attempts

C1 - FPA Limitations for Real-time software

C2 - Identification of previous attempts

C3 - Analysis of strengths & weaknesses

C4 - Testing of «3D» Proposal & feedback

C5 - Re-design requirements

C1 - FPA Limitations for Real-time Software

Does not capture well real-time functional characteristics:

- Large number of sub-processes
- Many transient data
- Many control functions

C1 - FPA Limitations for Real-time Software

FPA limitations recognized by the research & practitioners communities:

- Conte (1986)
- Jones (1988-...)
- Symons (1988)
- Ince (1991)
- Grady (1992)
- Whithmire (1992)
- Kan (1993)
- Hetzel (1993)
- Murali (1997)
- etc.

C2 - Previous Attempts

- Whitmire *3D FP*
- Jones *Feature Points*
- Symons *Mark-II*
- Reifer *Asset-R*
- Mukhopadhyay-Kekre *Robots*
- IFPUG *Case Study 4 (draft)*

C2 - Previous Attempts

Different solutions to avoid current limitations of actual FPA rules

- Four types of solution:
 - ❖ Addition of new components (*Jones, Whitmire*)
 - ❖ Adjustment of the Function Point count (*Mukhopadhyay-Kekre and Reifer*)
 - ❖ Continuous adjustment tables (*Symons*)
 - ❖ *Orthodoxy (IFPUG - New Environment Committee)*
- Objectives met ?

C3 - 3D Testing: Key Contribution

- MIS software



- Real-time software



C3 - 3D Testing & Feedback

- Not precise enough for definitions of detailed rules for the Transformation new function type
- State-Transition Diagrams:
 - ❖ Not available at project field sites
 - ❖ Management said it was always done

but

 - ❖ in practice,it is not there!!!

C4 - Re-Design Requirements

The measurement method to be proposed had to be based on:

- Criteria No. 1:
 - ❖ The practitioners had to agree that functional size had been ADEQUATELY captured (measured) for real-time software
- Other measurement and strategic criteria ...

C4 - Re-Design Requirements

MEASUREMENT Criteria:

- Current practices of documenting functional user requirements
- Concepts & vocabulary understandable by practitioners in real-time software
- Procedures to ensure:
 - ❖ repetitiveness
 - ❖ ease of use
 - ❖ low additional effort
 - ❖ + **other ISO quality criteria for measurement methods**

C4 - Re-Design Requirements

STRATEGIC Criteria:

- Build on IFPUG contributions to the software measurement field
- Facilitate migration path for organizations with historical FP repositories
- Keep IFPUG relevant
- Alignment with ISO framework in-progress

D - Full Function Points (FFP): FPA Extension for Real-time Software

D1 - Key Concepts

D2 - Full Function Points (FFP)

D3 - FFP Procedure

D4 - Example

D5 - Why FFP is easier

D6 - FFP integrated approach

D1 - Key Concepts

Generic Process



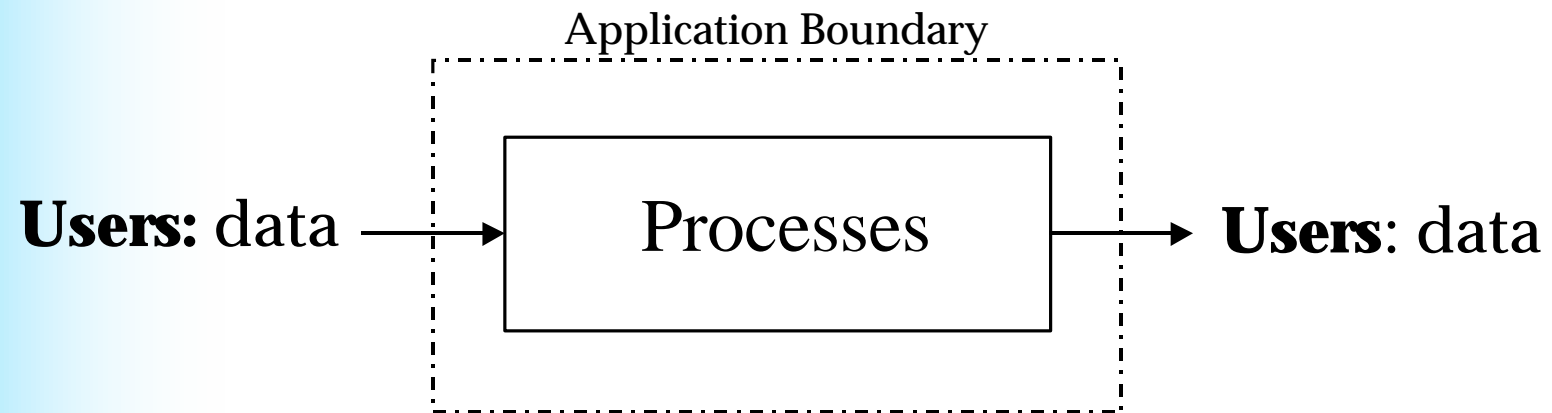
D1 - Key Concepts

Software Process



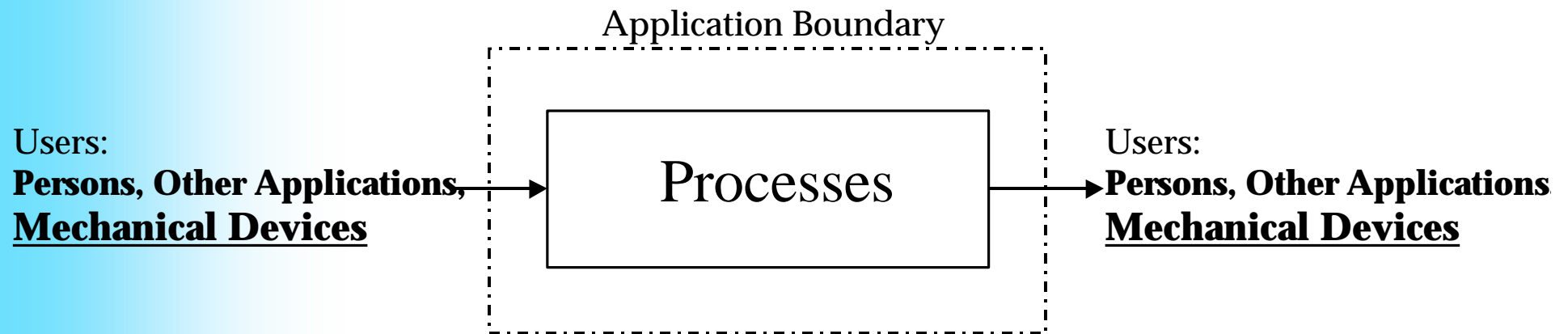
D1 - Key Concepts

Measured Software Processes



D1 - Key Concepts

Real-time Software Processes

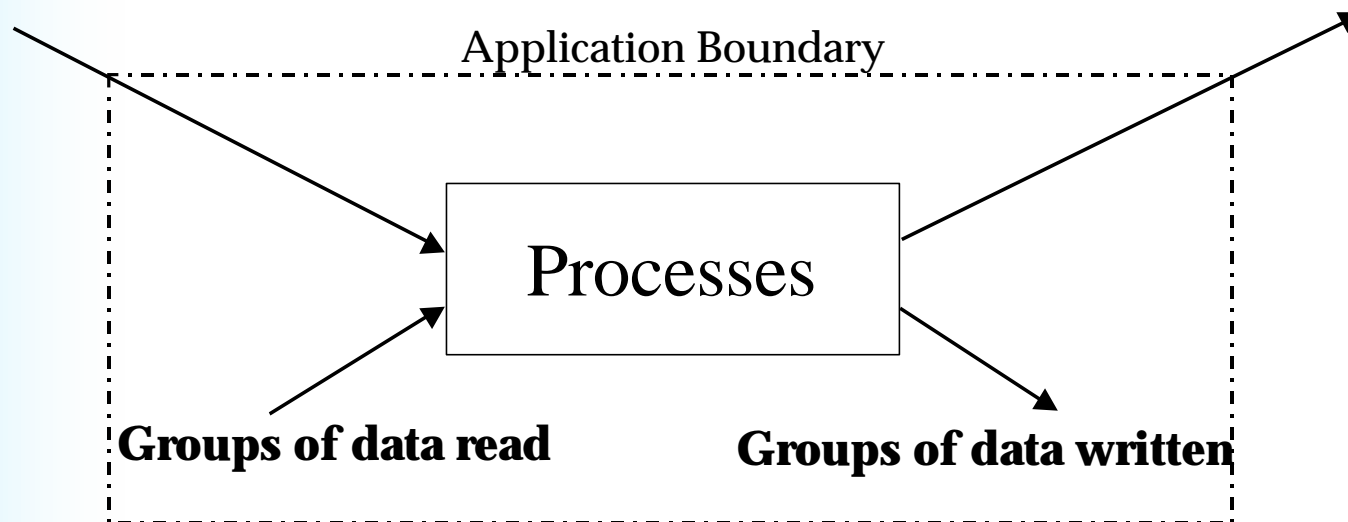


D2 - Full Function Points

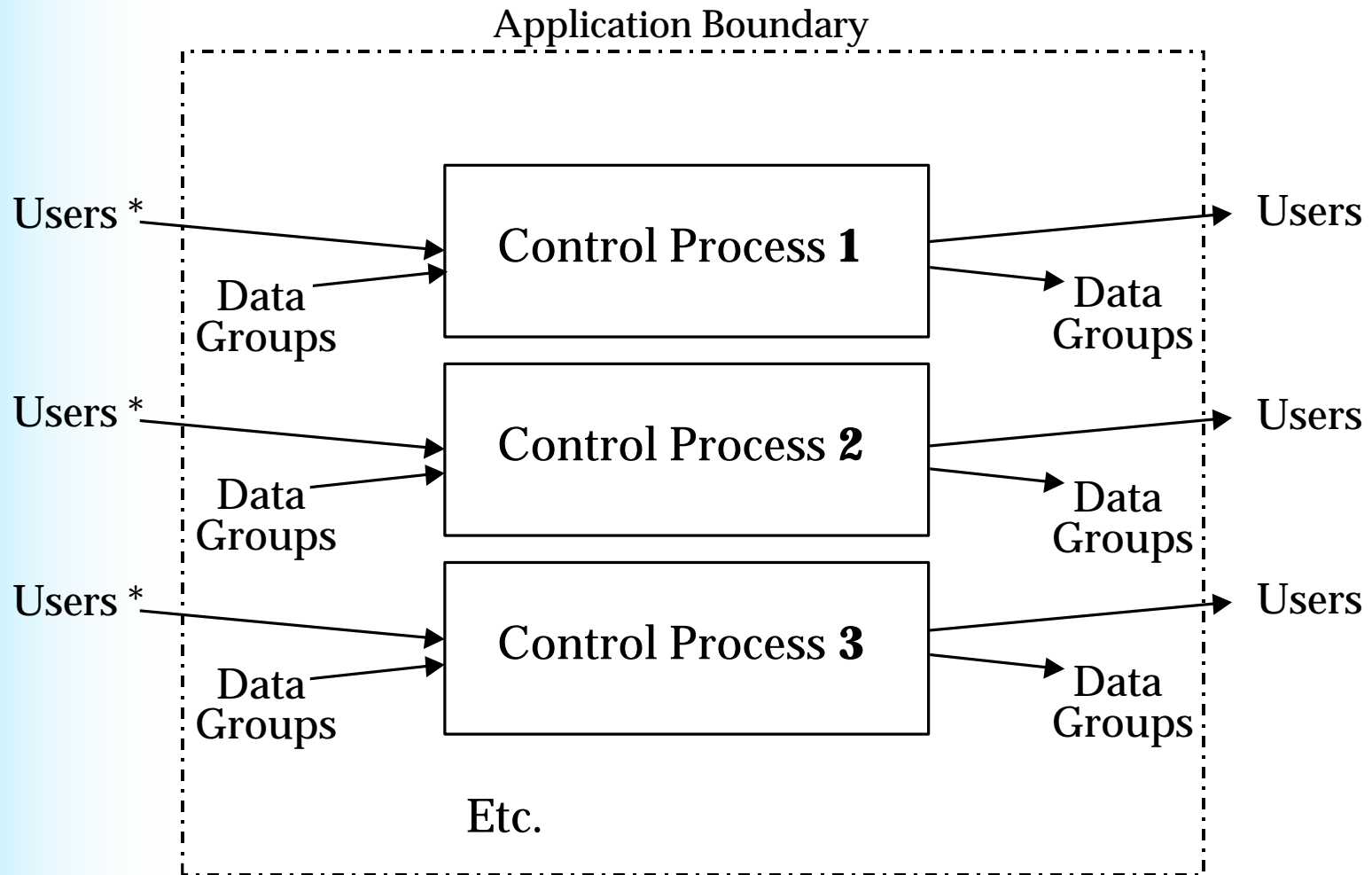
Real-Time Measurement Structure

Users: Persons, Other Applications,
Mechanical Devices

Users: Persons, Other Applications,
Mechanical Devices



D3 - FFP Procedure Processes



*: All processing associated with a unique trigger

D3 - FFP Procedure

Sub-processes

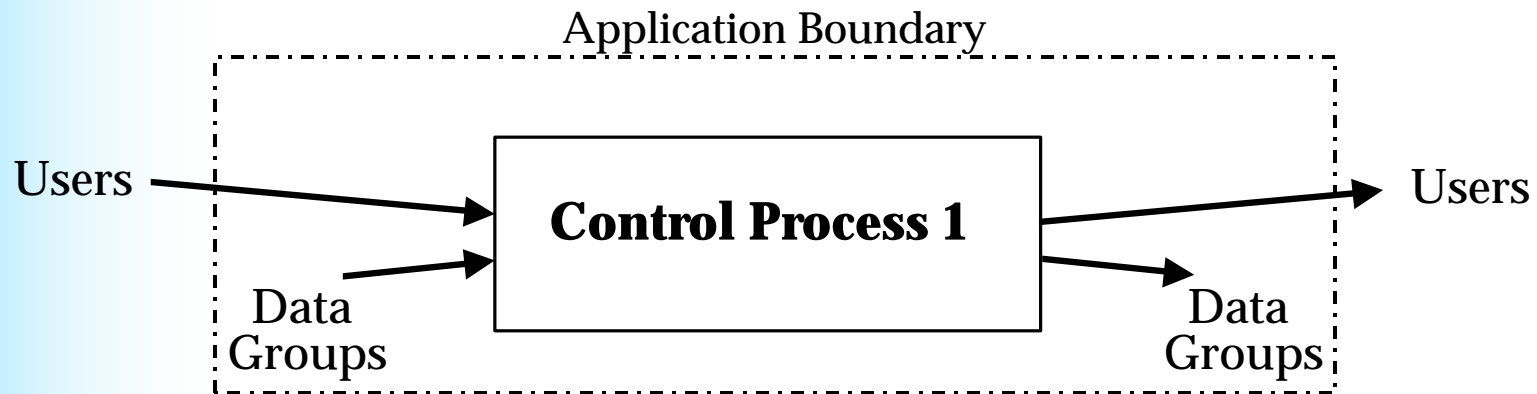


Figure 1

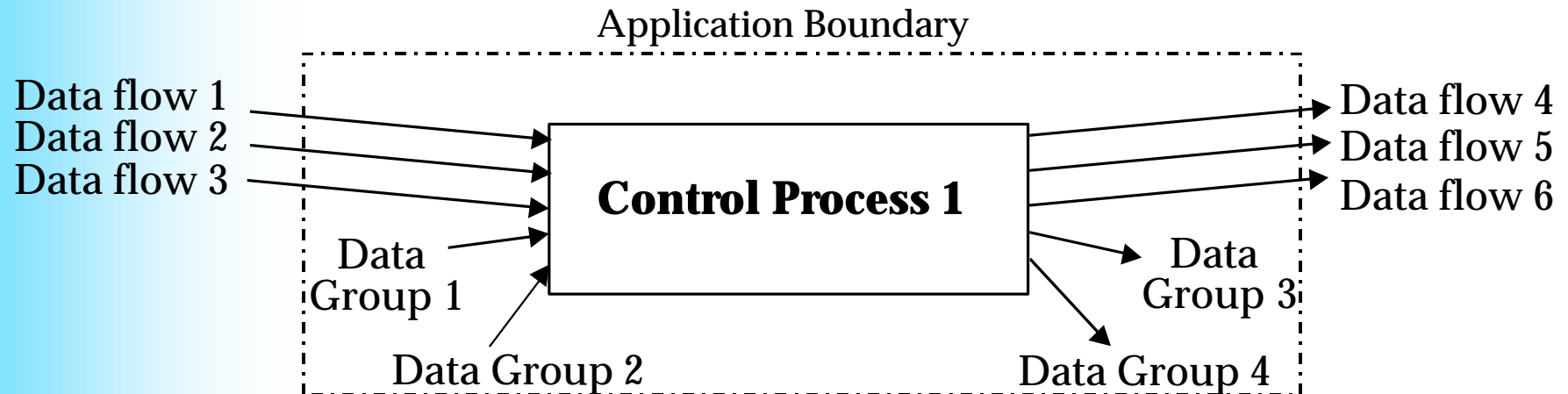


Figure 2

Note: in figure 2 each arrow is a sub-process.

D3 - FFP Procedure

Points Assignment

- Points are assigned at the **sub-process** level
- Weights were determined in order to be **aligned with IFPUG 4.0**
- Refer to the technical report for more details:

<http://www.lmagl.qc.ca/rtreport.pdf>

or

http://saturne.info.uqam.ca/Labo_Recherche/Lrgl/publi/treports/LRGL-1997-015.pdf

D3 - FFP rules

Entry Example

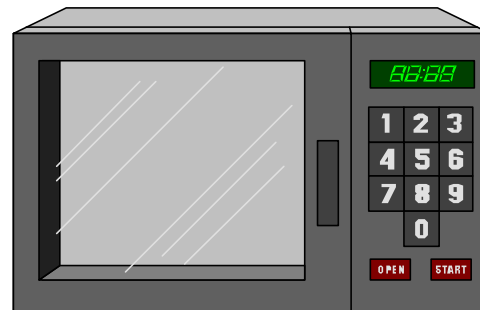
- The sub-process receives a group of control data from outside the application boundary.
- The sub-process is associated with only one group of data.
- The sub-process does not exit, read, or write data.
- The sub-process is unique: processing and data elements identified are different from other Entry of the same process.

D4 - Example

Requirement Specifications

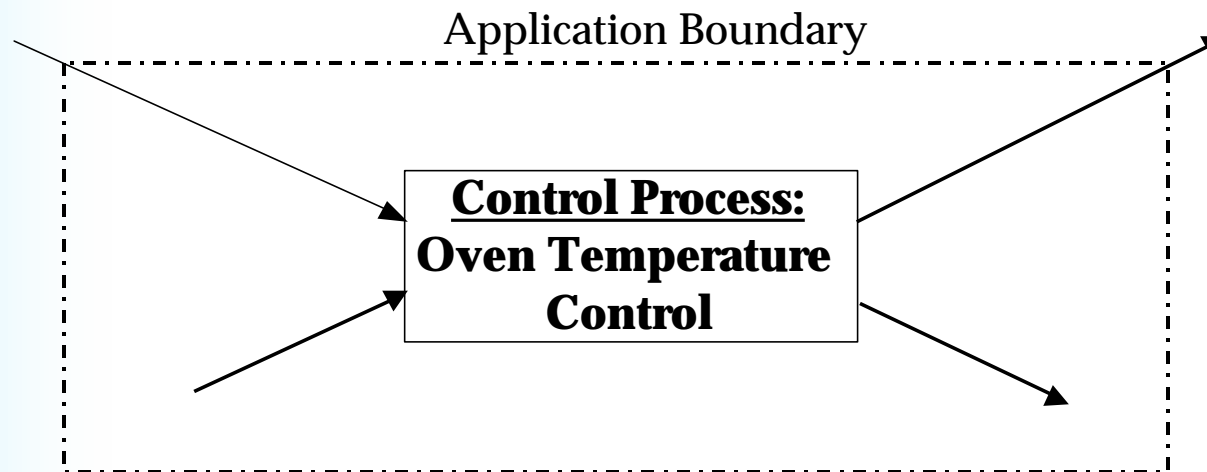
Oven Temperature Control

- 1) The Oven Temperature is **received** from a sensor
- 2) A message is **sent** to the oven heating element, turn on or turn off depending on Oven Temperature and **Desired Temperature**
- 3) A new entry is created in the **Message Log** (for diagnostic purposes)



D4 - Process Identification

All processing associated with a unique **trigger**:
“Temperature is received from the sensor”

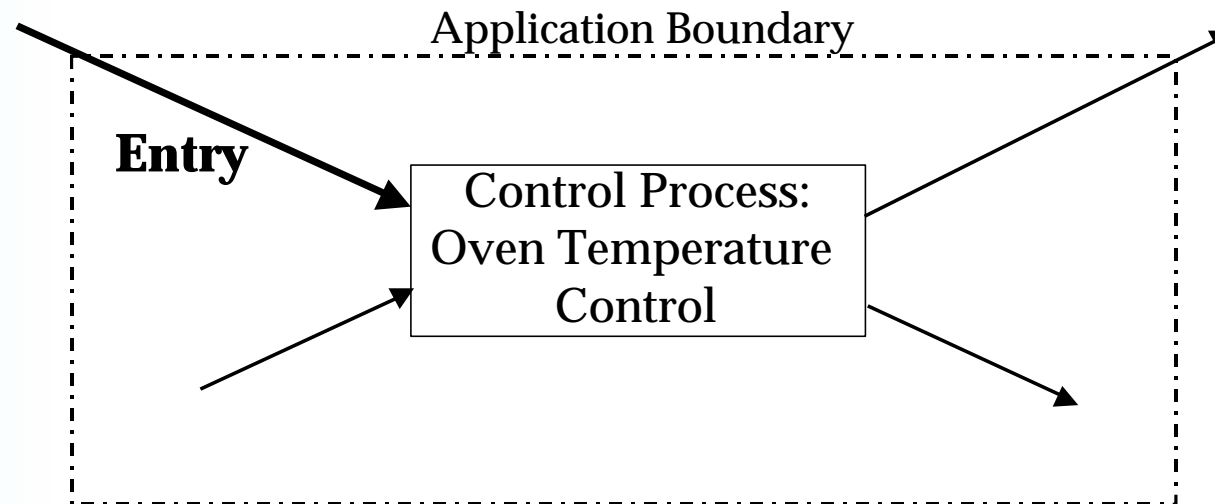


D4 - Sub-process Identification

Requirement Specification 1:

“The Oven Temperature is **received** from a sensor”

**The Oven Temperature is
received from a sensor**



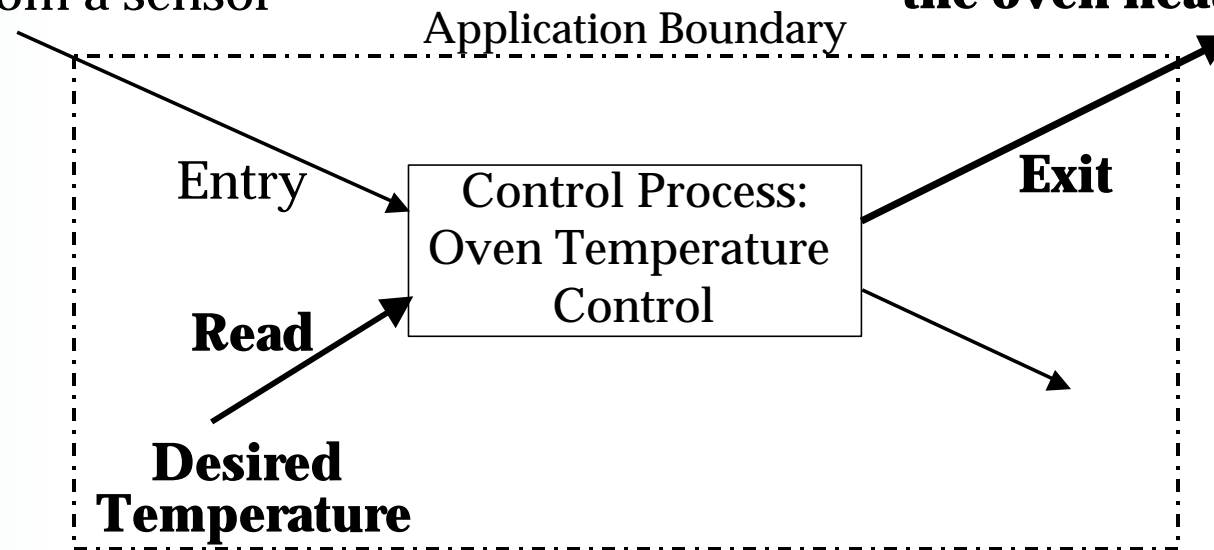
D4 - Sub-process Identification

Requirement Specification 2:

“A message is sent to the oven heating element, turn on or turn off depending on Oven Temperature and **Desired Temperature”**

The Oven Temperature
is received from a sensor

**A message is sent to
the oven heating element**



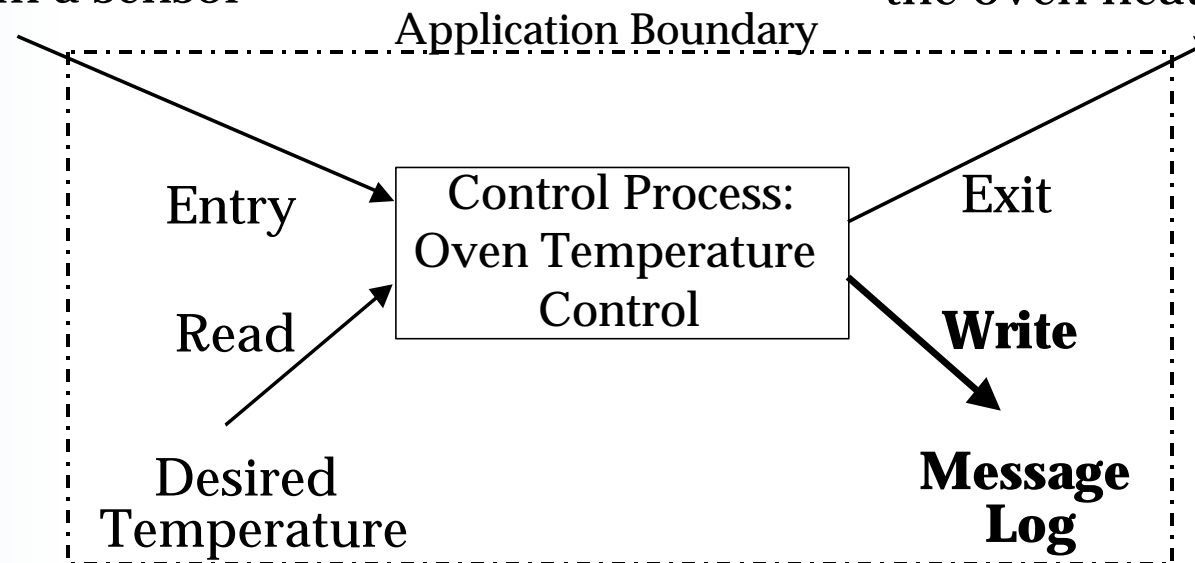
D4 - Sub-process Identification

Requirement Specification 3:

“A new entry is created in the **Message Log**”

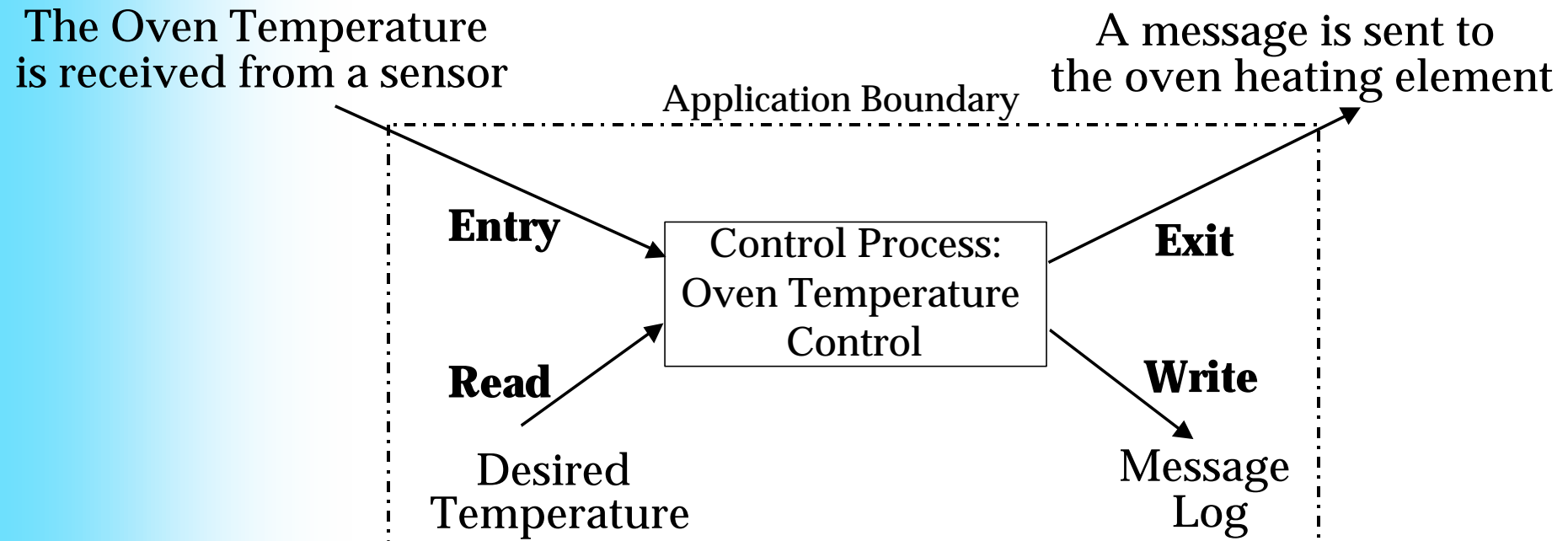
The Oven Temperature
is received from a sensor

A message is sent to
the oven heating element



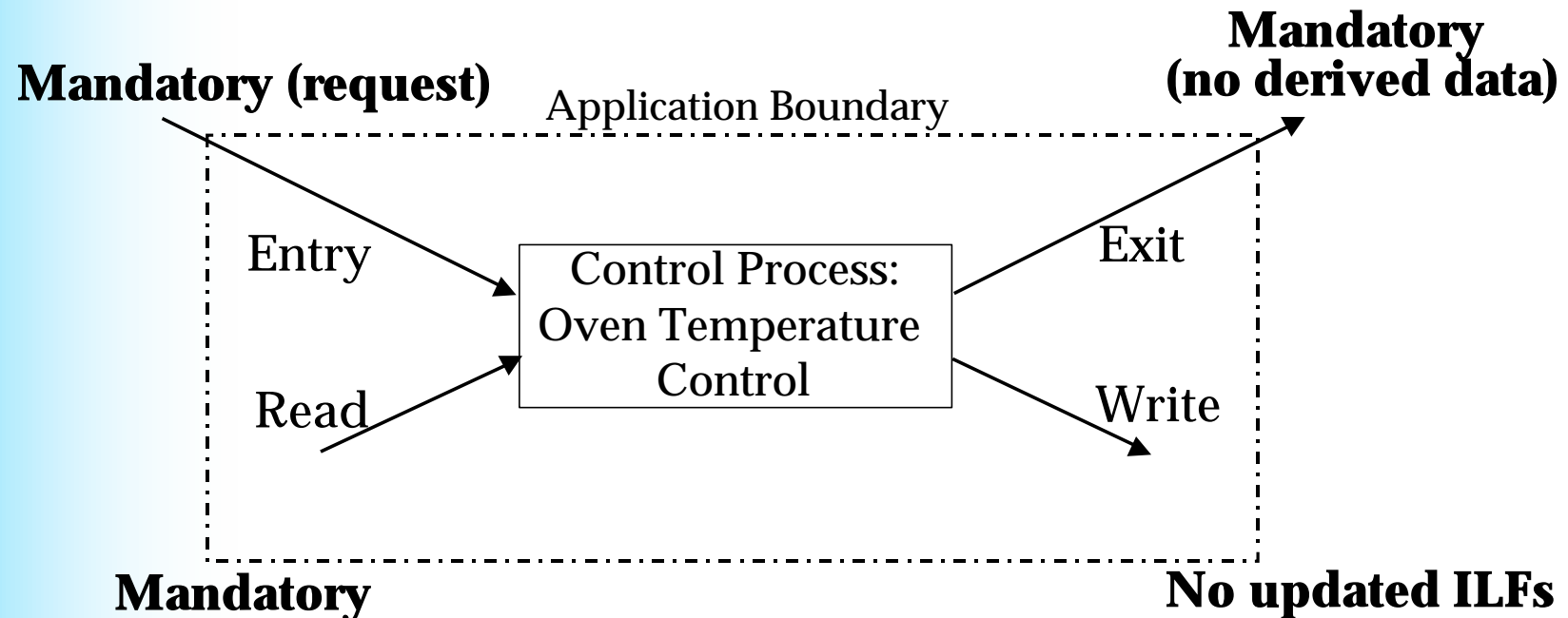
D4 - Sub-process Identification

Summary of Control Function Types: a single process



D5 - Why FFP is easier

Mapping of **External Inquiries (EQ)** on FFP



EQ additional rules:

- The process is the smallest unit of activity that is meaningful to the end user in the business
- The process is self-contained and leaves the business of the application in a consistent state
- For the identified process, one of the following two rules must apply:
 - Processing logic is unique from other external inputs for the application
 - The data elements identified are different from other external inputs for the application

D2 - Full Function Points

A Model for MIS and Real-time Software

*SOFTWARE = MIS Processes + **Control Processes***

SOFTWARE SIZE = IFPUG 4.0 + **Control Function Points***

* *note:* IFPUG 4.0 - a few control information rules

E - Industry Field Tests in Japan

E1 - Short Presentation of JSR

E2 - JSR Objectives

E3 - Field Tests

E4 - Conclusion

E1 - Short Presentation of JSR

- **JSR: JECS SYSTEM RESEARCH CO., LTD.**
 - ❖ Located in Tokyo, Japan
 - ❖ Moristugu Araki, Managing Director
- **JSR current business: To develop real-time software for the automotive industry in Japan including fuel injection systems**

E2 - JSR Objectives

- Our goal:
 - 1) To improve relationship with customer
 - 2) Continuous improvement in certain internal management issues:
 - ❖ Quality
 - ❖ Cost
 - ❖ Delivery Rate
- Our target:

To establish and implement our counting method for real-time software functional size at our corporation at the latest in 1998

E3 - Field tests: DESCRIPTION

Activities we have conducted:

- Study of IFPUG FPA, 3D Function Point (1994-1995)
- Counting trial (1996)
 - ❖ Count functional size of JSR Case Study model (Rice Cooker) by «IFPUG approach» and «3D approach»
 - ❖ Visit and Study from 3 Authorities, Dr.Abran, Mr. Garmus and Mr. Whitmire, based on the count results
- FFP Field Testing (1997)

E3 - Field tests: STRATEGY

JSR quality criteria for the method to be selected

- 1) To measure WELL real-time software functional size at our corporation
- 2) Easy count and same counting result by different counters
- 3) To implement in 1998

E3 - Field tests: STEPS

- 1) Study and count software functional size of JSR Case study model (Rice Cooker) and actual in-house samples with «FFP» based on documents sent by Dr. Abran (Jan-Apr 1997)
- 2) Visit and Study from Dr. Abran, Mr. Bourque and Mrs. Maya in UQAM based on the count results (May 1997)
- 3) Expand the testing (June 1997)

E3 - Field tests: SAMPLES

Characteristics of our real-time software

- Few Inputs and Outputs
- Few Files in boundary
- Some processes have a few sub-processes and some processes have a lot of sub-processes to control objects. Control dimension is essential for our real-time system
- Our system contains many very simple formulas because of the very tight cycle time constraints. Function dimension is not so essential as control dimension

E3 - Field tests: SAMPLES

Samples we counted

- Case study model (Rice Cooker): A few sub-processes sample
- Small engine control system: A few sub-processes sample
- Large engine control system: A lot of sub-processes sample

E3 - Field tests: Samples SIZE

Counting results

- Small samples (including Rice Cooker) = about 20 points
- Large sample = about 400 points

E3 - Field tests: RESULTS

Criteria 1: To measure WELL real-time software functional size at our corporation

- FFP can count very well the control functional size of our software
- The level of counting points might be acceptable judging from our experience
- We have been looking for the method to meet this criteria. FFP is very close to what we want

E3 - Field Tests: RESULTS

Criteria 1: To measure WELL real-time software functional size at our corporation (cont'd)

- In the large sample, FFP can count **79 sub-processes out of 81** which we expect to be counted. FFP cannot count 2 sub-processes, because FFP does not count such kind of function that contains only internal algorithms. It might be a weakness of this approach.

However, fortunately we do not have so many such sub-processes.

In this case, the count cover rate is **97%**.

This is sufficient for our practical use.

E3 - Field tests: RESULTS

Criteria 2: Easy count and same counting result by different counters

- Concepts and counting procedures in the FFP counting manual were relatively clear and easy to understand by us. Actually we made some mistakes. But they were not serious.
- We think FFP can sufficiently meet this criteria.

Criteria 3: To implement in 1998

- We have finished original study and are expanding it. Fortunately we have got good results so far.
- We think that we can probably establish our method based on FFP and hopefully implement it at least for our internal use.

E3 - Field tests: OBSERVATIONS

About IFPUG approach

- IFPUG approach is excellent especially for MIS software. In Japan, it is becoming popular. Members of Japan Function Point Users Group are 140 this summer. It is twice of last summer.
- IFPUG approach, however, might be difficult to measure well our Real-time software environment.
- In our study, IFPUG approach could count almost same points as FFP in SMALL samples, but could not count so many points as FFP in the LARGE sample. It means that IFPUG points could not scale up to large counts as well as we expected, at least in our environment.

E3 - Field tests: OBSERVATIONS

About 3D approach

- 3D approach seems attractive for us. It measures not only data dimension and control dimension but also formula dimension (function dimension). In our study, points of this approach were larger than FFP. And it scaled up as well as FFP.
- However, in case of our actual complicated systems, it was not easy to count especially control dimension by using the State-Transition Diagrams. Other difficulty might come from lack of sufficient information to explain how to count. For these reasons we could not consider to utilize and implement it by 1998 in our corporation.

E4 - CONCLUSION

- At least at this moment, we think that FFP might be most preferable for us (through our field tests).

By great and kind guidance and support of many persons, we could reach at this conclusion . We would like to thank sincerely all of those persons.

F - Conclusion

F1 - Project status

F2 - Research next steps

F3 - Observations

F4 - Closing remarks

F1 - Status: Re-design Criteria Met

Criteria No. 1:

- Practitioners agree that Functional Size ADEQUATELY captured for real-time software
- Verification method: field tests feedback
(Further verification required with methods yet to be developed in the field of software metrics!)

F1 - Status: Re-design Criteria Met

- Measurement criteria:
 - ❖ Current practices of documenting
 - ❖ Concepts & vocabulary in real-time software
 - ❖ Procedures to ensure: repetitiveness, ease of use, + **other ISO quality criteria**
- Strategic criteria:
 - ❖ Build on IFPUG contributions
 - ❖ Facilitate migration path
 - ❖ Keep IFPUG relevant
 - ❖ Alignment with ISO framework in-progress

F1 - Status: Deliverables

- FFP: Documented and in the public domain
- A Counting Practices Manual for FFP

F2 - Research: Next Steps

Phase A: Design & Measurement Procedures

- Industry partners tests conclusive

Phase B:

- Degree of repetitiveness
 - ❖ IFPUG/M.I.T.-type studies required
- Usefulness of FFP in productivity, estimation and quality models
 - ❖ Research requirements: FFP measurement of completed projects WITH effort data in semi-controlled environments at industrial sites.

F3 - Observations

Other organizations: YOUR call and YOUR strategy

- Pioneers
- Main stream
- Late adopters

F3 - Observations: IFPUG

Landmarks:

- Albrecht 79:
 - ❖ Visionary
 - ❖ Most measurement steps
 - ❖ A single context and a small subset of projects
- IFPUG mid 80 's:
 - ❖ Measurement process: Detailed measurement procedures to ensure consistency across contexts, technologies and time but within the same single MIS domain

F3 - Observations: IFPUG

Landmarks: (cont'd)

- IFPUG early 90 's:

- ❖ Accuracy: certification programs

- IFPUG late 90 's: ??????

Which strategy will IFPUG take to gain relevance
OUTSIDE of its traditional MIS domain:

- ❖ Orthodoxy

or

- ❖ Leadership in functional size measurement methods
and their use for management purposes

F4 - Closing Remarks

- The problem of the relevance of measuring real-time software with Function Points has been known for at least 10 years!
- Who has put money and resources on the table to contribute to the development of a solution?

F4 - Closing Remarks

Thanks to the consortium partners:

- NORTEL, JSR, BELL and Hydro-Québec for their:
 - ❖ \$\$\$\$
 - ❖ Time
 - ❖ Access to their software
 - ❖ Access to their staff
 - ❖ Their most valuable feedback

F4 - Closing Remarks

Thanks to the consortium partners (Cont'd)

- SELAM for their:
 - ❖ Project initiation and kick off
 - ❖ Function Point expertise
 - ❖ Effort and dedication

F4 - Closing Remarks

Thanks to the consortium partners (Cont'd)

- *Université du Québec à Montréal* team for their:
 - ❖ Research infrastructure
 - ❖ Leadership in research on functional size measurement

F5- Questions Period



For more information

A. Abran: abran.alain@uqam.ca

M. Araki: ldk02326@niftyserve.or.jp

J.M. Desharnais: desharnais.jean-marc@uqam.ca

D. St-Pierre: Denis.St-Pierre@crim.ca

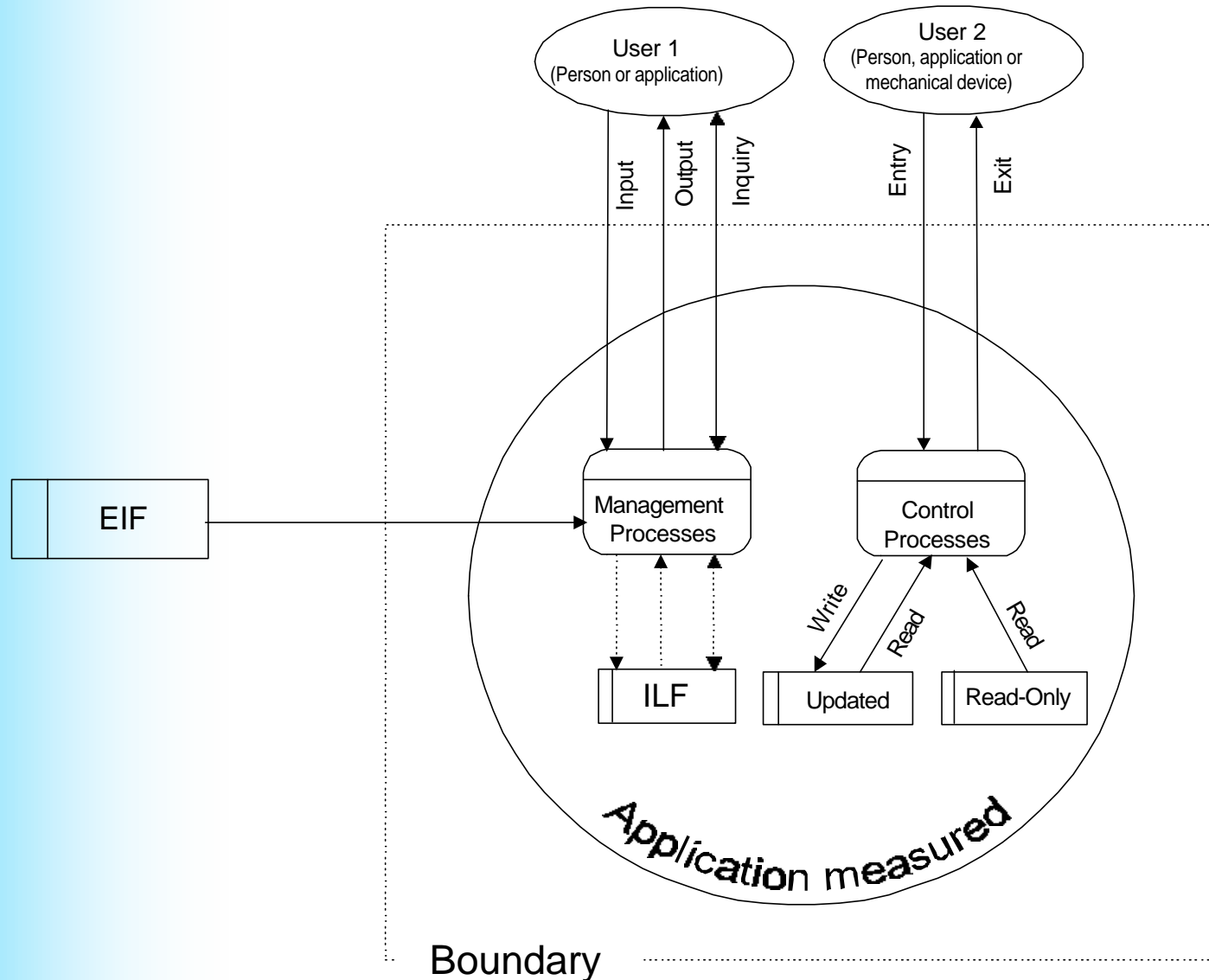
Web sites:

http://saturne.info.uqam.ca/Labo_Recherche/lrgl.html

<http://www.lmagl.qc.ca>

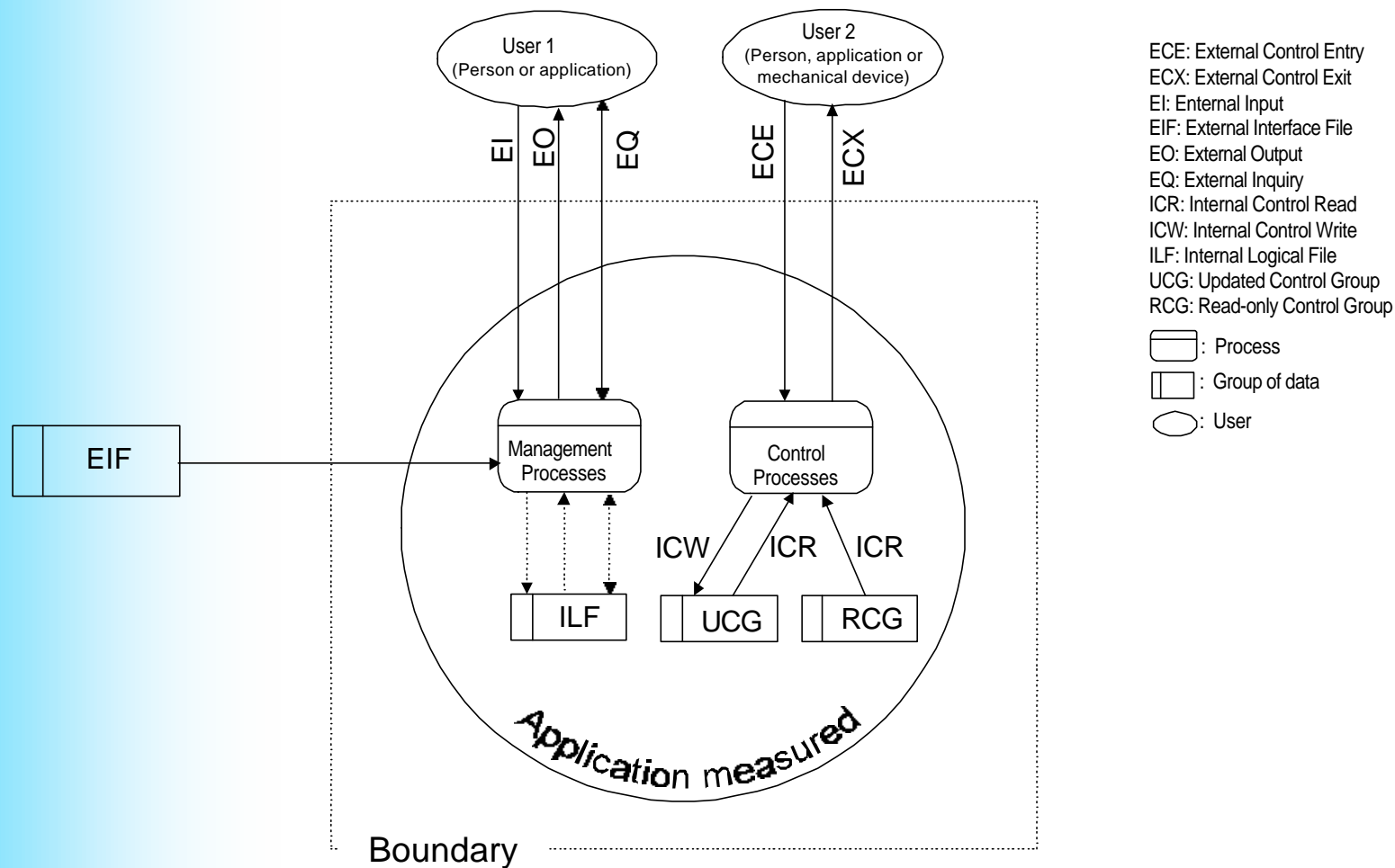
D6 - FFP Integrated Approach

FFP = *IFPUG 4.0* + *Control Function Points*



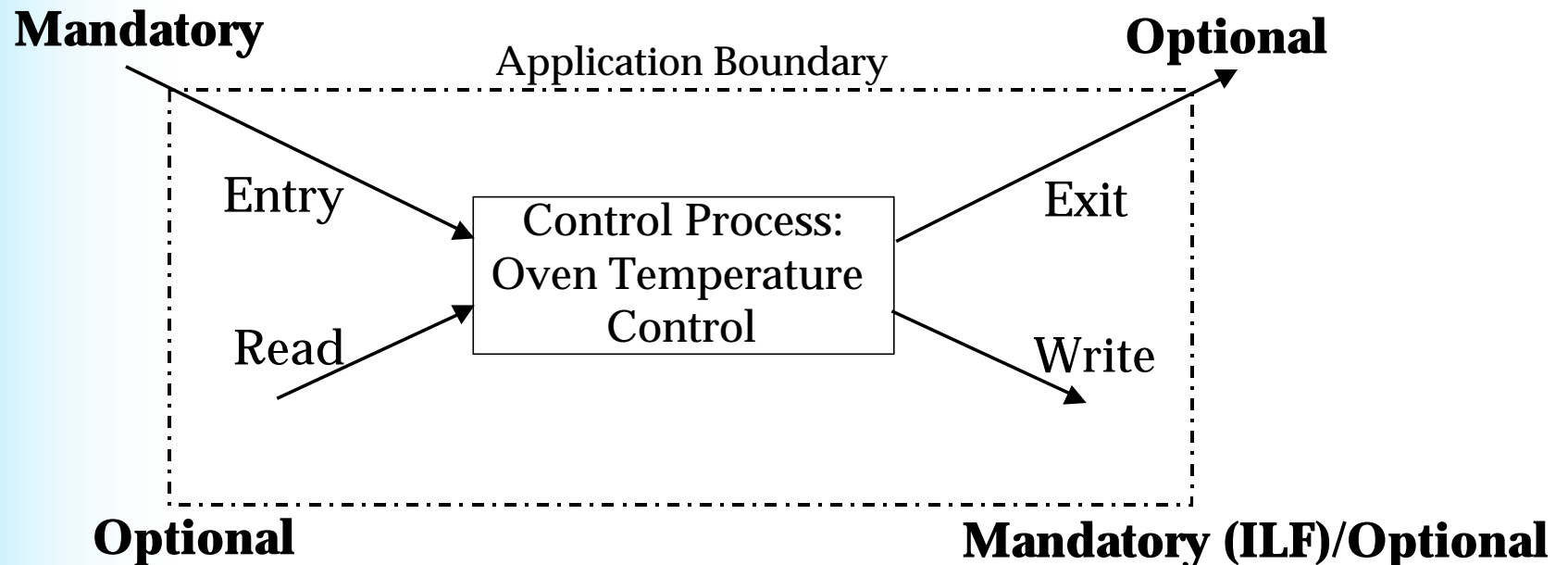
D6 - FFP Integrated Approach

New Function types



D5 - Why FFP is easier

Mapping of **External Inputs** (EI) on FFP

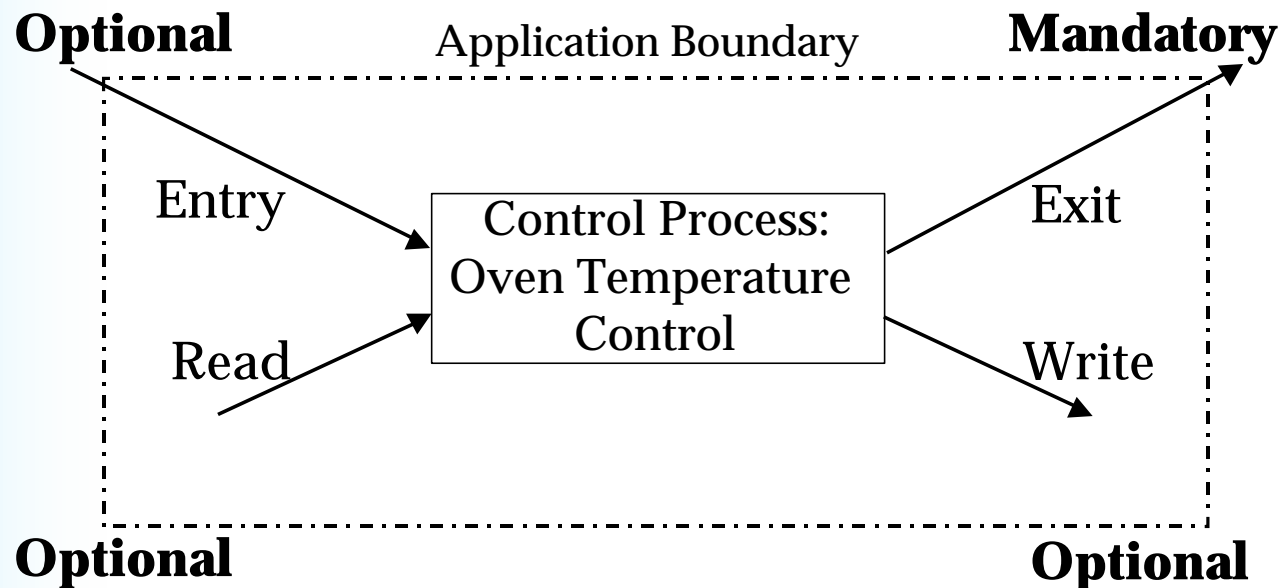


EI additional rules:

- The process is the smallest unit of activity that is meaningful to the end user in the business
- The process is self-contained and leaves the business of the application in a consistent state
- For the identified process, one of the following two rules must apply:
 - Processing logic is unique from other external inputs for the application
 - The data elements identified are different from other external inputs for the application

D5 - Why FFP is easier

Mapping of **External Outputs** (EO) on FFP



EO additional rules:

- The process is the smallest unit of activity that is meaningful to the end user in the business
- The process is self-contained and leaves the business of the application in a consistent state
- For the identified process, one of the following two rules must apply:
 - Processing logic is unique from other external inputs for the application
 - The data elements identified are different from other external inputs for the application