Measuring the Functional Size of Real-Time Software

Marcela Maya, Alain Abran, Serge Oligny, Denis St-Pierre
Jean-Marc Desharnais

Agenda

- **Motivation and context**
- **How FFP was developed**
  - Project Goals and Deliverables
  - Project Structure
- **Key concepts**
  - Function Point Analysis (FPA)
  - Full Function Points (FFP)
- **Major differences between FPA and FFP**
- **Field testing FFP**
- **Conclusion**

Motivation and context

✦ **Functional characteristics of real-time software are not well captured by Function Point Analysis (FPA).**

(Jones, 1991; Whitmire 1992; Galea, 1995)

✦ **Generally speaking, FPA counts on real-time systems tends to be low.**

✦ **Therefore FPA is not perceived as an adequate functional size measure for real-time systems.**

✦ **There is no FPA equivalent technique for the real-time domain.**

Motivation and context

✦ **Previous attempts to adapt FPA to real-time software:**

  ↳ Mark II (Symons, 1988)

  ↳ Asset-R (Reifer, 1990)

  ↳ Feature Points (Jones, 1991)

  ↳ Application Features (Mukhopadhyay and Kekre, 1992)

  ↳ 3D FP (Whitmire, 1992)

  ↳ IFPUG Case Study 4 - Draft version (IFPUG, 1997)

✦ **None of these approaches has succeeded in gaining wide market acceptance.**

How FFP was developed...

✦ **FFP project goals**

   ↳ **Retain the actual FPA quality characteristics from a measurement perspective:**

      ⮑ **Relevance** (adequate from the users perspective)

      ⮑ **Instrumentation** (counting practices and procedures)

      ⮑ **Practicality and applicability** (based on actual S.E. practices)

      ⮑ **Transferability** (to a standard setting body)

   ↳ **Adapt FPA to take into account the specific functional characteristics of real-time software**

   ↳ **Align, as much as possible, with the standard FPA (IFPUG, 1994).**

How FFP was developed...

- ✦ **FFP project deliverables**
  - ⇨ **Short term**
    - ➲ Detailed procedures and rules
    - ➲ Field test

  - ⇨ **Long term**
    - ➲ Productivity Model
    - ➲ Technology transfer session
    - ➲ Contribution to IFPUG Standards
    - ➲ Contribution to ISO Standards

How FFP was developed...

Project organization

UQÀM
SEMRL

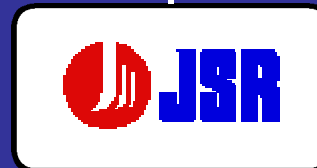SELAM
LMAGL

↪ **Concepts development**

Industrial partners

Bell

N⊘RTEL
NORTHERN TELECOM

JSR

Hydro Québec

Field tests

↪ **Quebec**
↪ **Montreal**

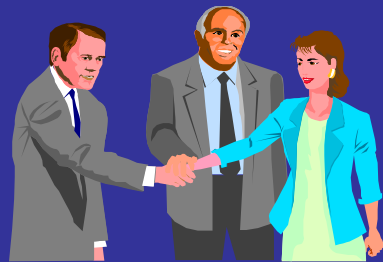↪ **Dallas, TX**
↪ **Toronto**
↪ **Ottawa**

↪ **Tokyo, Japan**

↪**Montreal**

Key concepts: FPA

Measure of the functional size of a software application from a users perspective
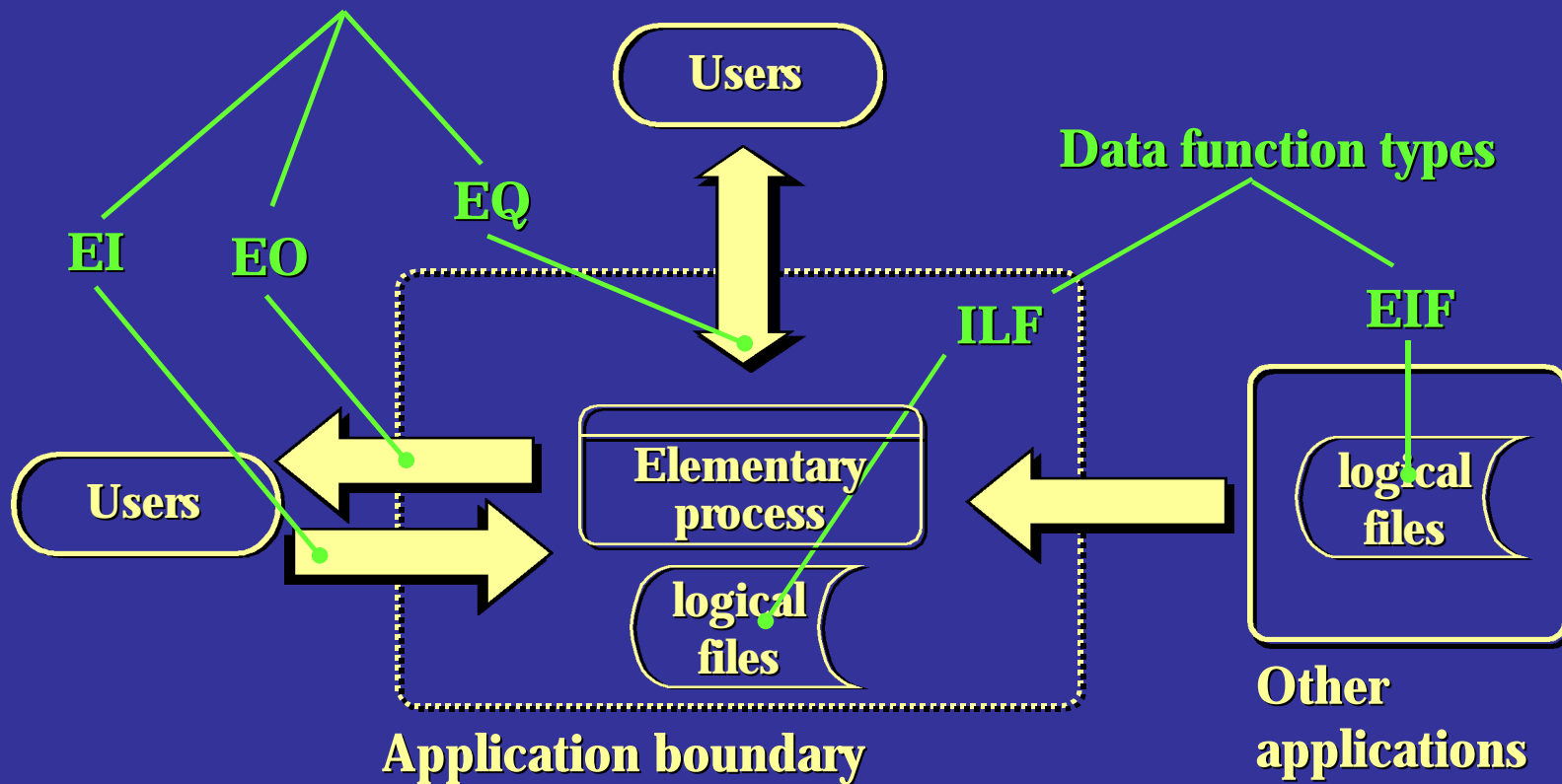
↪ Data entry
↪ Production of reports
↪ Data storage
↪ Inquiry on data
↪ Interactions with other systems

↪ Inputs
↪ Outputs
↪ Internal logical files
↪ Inquiries
↪ External Interface files

Key concepts: FPA

Transactional function types

Data function types

Users

EQ

EI   EO

ILF

EIF

Users

Elementary process

logical files

logical files
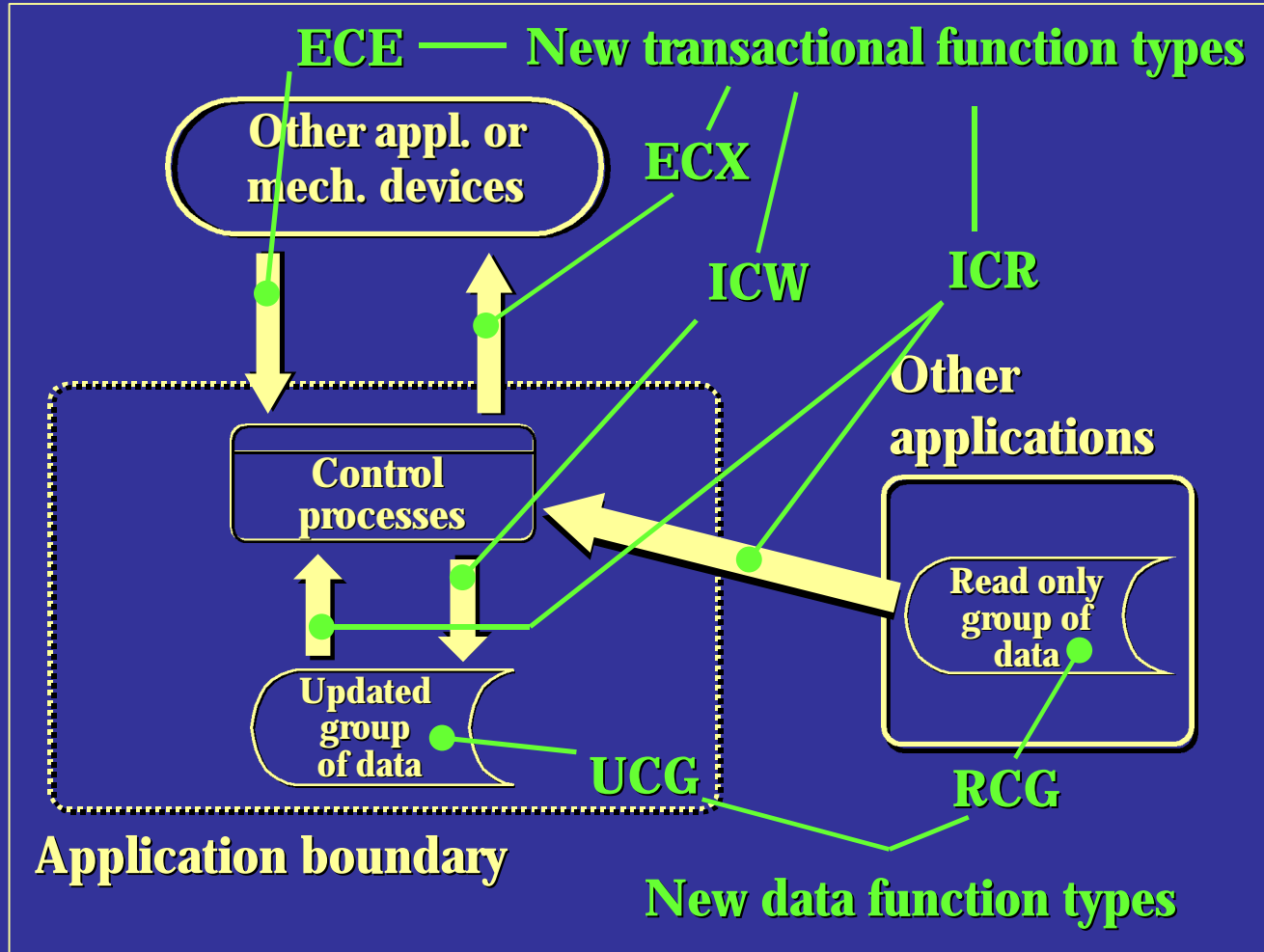
Application boundary

Other applications

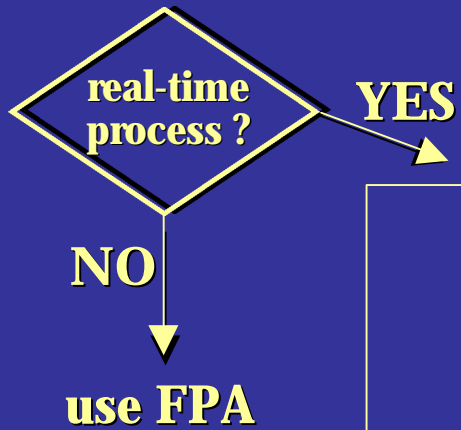# Key concepts: FPA

✦ <u>Elementary processes</u> are the smallest units of activity that is meaningful to the business end users.

⇨ Data function types (ILF, EIF) are maintained via elementary processes.

⇨ Transactional function types (EI, EO, EQ) are the elementary processes.

FPA: caveats when counting real-time systems

- ➲ Distinction between EI, EO and EQ is blurred in real-time system,

- ➲ The number of single occurrence data is often very significant and it is not considered by FPA,

- ➲ The number of sub-processes within real-time processes might vary a lot,

- ➲ The number of control data is often very significant.

- ➫ FFP introduces new concepts to take into account these characteristics.

Key concepts: FFP

real-time process ? — **YES**

NO

use FPA

ECE —— New transactional function types

Other appl. or mech. devices

ECX

ICW

ICR

Other applications

Control processes

Read only group of data

Updated group of data

UCG

RCG

Application boundary

New data function types

Major differences between FPA and FFP

FPA

Elementary process

|

is classified as

input   output   inquiry

Elem. Proc. = input or
              output or
              inquiry

FFP

Control processes

|

is composed of

entries exits   reads   writes

Ctl. Sub-proc. = entries and
                 exits   and
                 reads   and
                 writes

Field testing FFP

↳ **Three real-time applications were measured using FFP and FPA between December 1996 and March 1997,**

(telecommunications and power supply)

↳ **Small application or a self-contained portion of a medium or large application, (± 25.000 LOC)**

↳ **Each counting session lasted two full days,**

↳ **At least three people participated in the counting sessions: an application specialist and two FFP experts,**

↳ **A fourth field test was conducted by one of the project's industrial partners without the assistance of the FFP specialists (using only FFP documentation)**

Field testing FFP: selected results

FPA	Application A		Application B		Application C	
	Occ.	Points	Occ.	Points	Occ.	Points
⇨ Inputs	40	202	6	21	15	50
⇨ Outputs	2	14	2	11	17	73
⇨ Inquiry	12	40	1	6	0	0
TOTAL	54	256	9	38	32	123
FFP						
⇨ Entries	123	123	10	10	67	69
⇨ Exits	93	97	8	10	136	139
⇨ Reads	395	403	14	18	100	103
⇨ Writes	142	154	8	8	165	168
TOTAL	753	777	40	46	468	479

Field testing FFP: comments

- FFP generate larger counts than FPA,

- The number of sub-processes of a real-time process varies a lot: some embedded only 3 sub-processes while others embedded more than 50 sub-processes,

- FFP and FPA counting efforts are similar,

- According to application specialists FFP offered them a more adequate measure of the functional size of their applications than FPA,

- Concepts, counting procedures and rules are clear and detailed enough to enable different individuals to come up with relatively similar results,

- They are based on current practices as to what is currently and effectively being documented in real-time software development.

Conclusion

⇨ **Practitioners agree that FFP ADEQUATELY capture the functional size of their for real-time software applications,**

⇨ **More field-testing is needed to provide feedback and improve the approach as well as the counting procedures and rules,**

⇨ **More field-testing will also bring enough empirical data to support the development of meaningful productivity and estimation models.**

Acknowledgements

We want to thank Nortel, Bell Canada, Hydro-Québec and JECS System Research (Japan) for providing project funds, industrial data and valuable feedback from real-time software practitioners.

http://saturne.info.uqam.ca/Labo_Recherche/Lrgl/ffp.htm

http://www.lmagl.qc.ca

????

References

Galea, S., "The Boeing Company: 3D Function Point Extensions, V2.0, Release 1.0", Seattle, WA: Boeing Information and Support Services, Research and Technology Software Engineering, June 1995.

IFPUG, "Function Point Counting Practices Manual, Release 4.0", International Function Point Users Group - IFPUG, Westerville, Ohio, 1994.

IFPUG, "IFPUG Case Study 4 (Draft)", International Function Point Users Group - IFPUG, Westerville, Ohio, 1997.

Jones, C., "Applied Software Measurement - Assuring Productivity and Quality, McGraw-Hill, New York, 1991, 493 pages.

Mukhopadhyay, T. and Kekre, S., "Software Effort Models for Early Estimation of Process Control Applications", IEEE Transactions on Software Engineering, Vol. 18, No. 10, October 1992, pp. 915-924.

Reifer, D. J., "Asset-R: A Function Point Sizing Tool for Scientific and Real-Time Systems", Journal of Systems and Software, Vol. 11, No. 3, March 1990, pp. 159-171.

St-Pierre, D., Maya, M., Abran, A. et Desharnais, J-M, "Full Function Points - Counting Practices Manual", Technical Report 1997-04, Software Engineering Management Research Laboratory, Université du Québec à Montréal (UQAM), September 1997, 49 pages.

Symons, C.R., "Function Point Analysis: Difficulties and Improvements", IEEE Transactions on Software Engineering, Vol. 14, No. 1, January 1988.

Whitmire, S. A., "3-D Function Points: Scientific and Real-Time Extensions to Function Points", Proceedings of the 1992 Pacific Northwest Software Quality Conference, Portland, OR, 1992.