An Estimation Improvement Program in a Software Organization

Vitor Alcântara Batista

Synergia – Engenharia de Software e Sistemas

Av. Antônio Carlos, 6627 – ICEx – sl. 3056

Belo Horizonte – MG – +55 (31) 3409-5855

www.synergia.dcc.ufmg.br

Synergia
Engenharia de Software e Sistemas

Summary

- Introduction
- Important concepts
- Methodology
- Diagnostics
- Proposed actions and results
- Conclusion

Introduction

- **Good estimates are essential for software development organizations:**
  - **Competitiveness.**
  - **Allow better project planning and monitoring.**
  - **Avoid excessive schedule pressure, proved to be destructive for projects [Nan & Harter, 2009].**

Introduction – Industry overview

- **Industry survey in 2008 [Yang et al., 2008]:**
  - **59% to 76% of software projects show effort overrun - estimation errors between 18% and 41%.**
  - **35% a 80% of software projects show schedule overrun - errors between 22% and 25%.**

Introduction - Motivation

- Although estimation in software has been widely studied for decades, so far it is not well solved.
- Main difficulties [McConnell, 2006], [Laird, 2006]:
  - Hope-based planning.
  - Confusion between goal and estimation.
  - Incomplete requirements.
  - Organizations don’t know their own capacity.
  - Unstable requirements (shooting at a moving target).

Introduction - Objectives

- **Performing a software estimation process improvement program (SPI) at Synergia:**
  - **Managers from Organization have claimed that projects have poor estimation accuracy.**

Important concepts

- At Synergia, estimation is performed in three distinct moments:
  - During project activation, to present a proposal for our customers.
  - Macro planning (iterations), when the project starts.
  - Detailed iteration planning, before each iteration begins.

Important concepts

- **Accuracy indicators**

$MRE_i = \frac{\lvert y_i - \hat{y}_i \rvert}{y_i}$ $MMRE = \sum_{i=1}^{n} MRE_i$	$BRE_i = \begin{cases} \frac{(\hat{y}_i - y_i)}{y_i} & se\ (\hat{y}_i - y_i) \geq 0 \\ \frac{(\hat{y}_i - y_i)}{\hat{y}_i} & se\ (\hat{y}_i - y_i) < 0 \end{cases}$ $MBRE = \frac{1}{n}\sum_{i=1}^{n} BRE_i$	**PRED(x) = % of errors bellow x**
$MER_i = \frac{\lvert y_i - \hat{y}_i \rvert}{\hat{y}_i}$ $MMER = \sum_{i=1}^{n} MER_i$	$IBRE_i = \begin{cases} \frac{(\hat{y}_i - y_i)}{y_i} & se\ (\hat{y}_i - y_i) < 0 \\ \frac{(\hat{y}_i - y_i)}{\hat{y}_i} & se\ (\hat{y}_i - y_i) \geq 0 \end{cases}$ $MIBRE = \frac{1}{n}\sum_{i=1}^{n} IBRE_i$	

$\hat{y}$ = estimated value, y = actual value

ProMOTe

- **Proprietary Improvement Process for Technical Organizations**
- **Based on SEI's IDEAL model.**

Initiating Phase

- Scope definition
- SPI program team definition
- Kick-off meeting

Diagnostics

- **Proposed actions after diagnostics:**
  1. **Define a process for proposal estimation**
  2. **Improve the macro planning process**
  3. **Define a method to count Function points directly on UML Models**
  4. **Define a process for detailed iteration planning**
  5. **Create a central repository for project data**

Action 1 – Estimation for proposals

- Adoption of COCOMO II [Boehm et al., 2000]:
  - Development of a Lines of Code (LOC) tool that conforms to SEI recommendations [Park et al., 1992].
  - Development of a parameters interpretation manual.
  - Including risk assessment in COCOMO modelling.

Action 1 – Estimation for proposals

- **Results (accuracy):**
  - **Effort (without model calibration)**

MMRE	MMER	MBRE	MIBRE	PRED(0,25)
319,50%	71,97%	319,50%	71,97%	0,00%

  - **Effort (with model calibration)**

MMRE	MMER	MBRE	MIBRE	PRED(0,25)
15,63%	14,44%	-0,31%	-1,51%	87,50%

  - **Schedule (with calibration)**

MMRE	MMER	MBRE	MIBRE	PRED(0,25)
43,34%	44,68%	0,17%	1,51%	37,50%

  - **Simulations used *leave-one-out* method.**

Action 2 – Estimation for macro planning (iterations)

- It makes no sense to elaborate a complete project WBS for all iterations:
  - Volatile requirements
  - Staff turnover
  - Large variation of effort and schedule in fine grained activities
- Solution: estimate only total effort, schedule and staff required for each iteration, based on its scope

Action 2 – Estimation for macro planning (iterations)

- Given required effort for each iteration and effort distribution from historical data, estimate:
  - Staff needed to achieve required schedule, or;
  - Likely schedule for available staff.

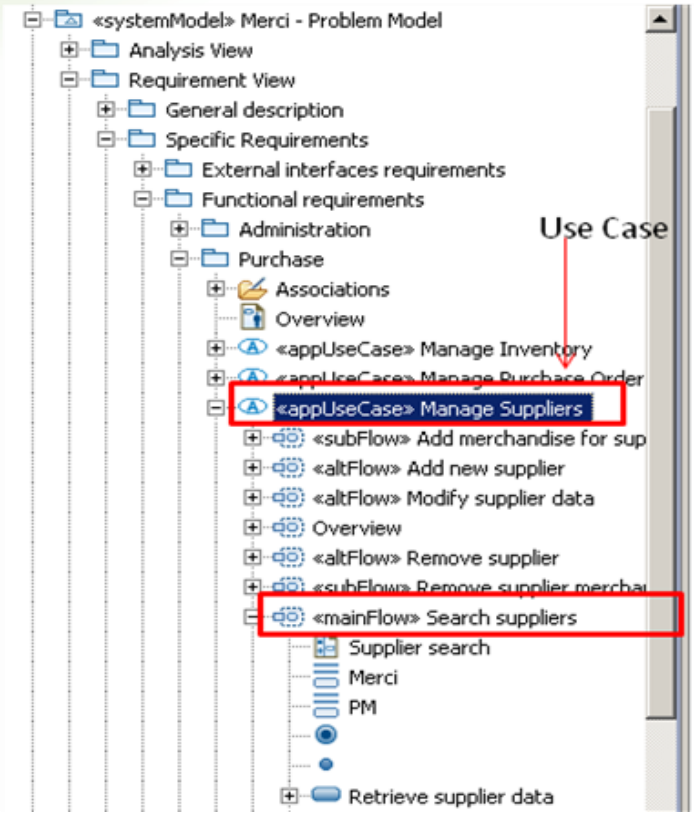
Action 3 – Count FP directly on UML Models

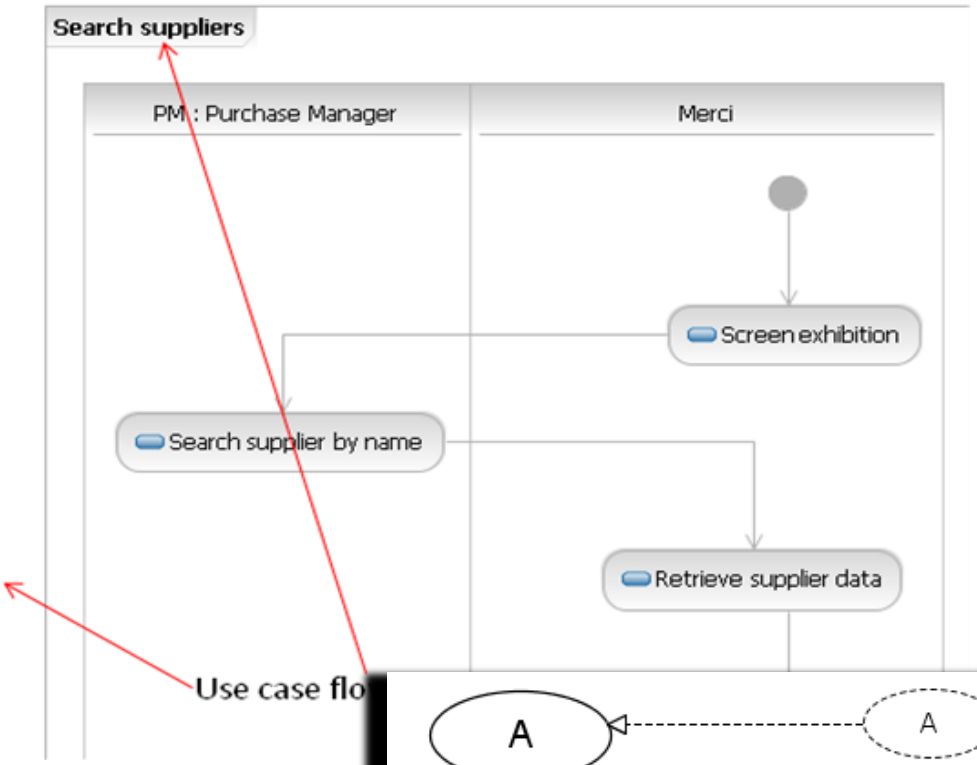
- Keep IFPUG counting practices as method.
- Development of a plug-in to modeling tool (Rational Software Modeler) to support counting using UML model:
  - Use of stereotypes to record FP counting data
  - User friendly interface to document FP count decisions (with error prevention)
  - Automatic model and counting validation using OCL constraints applied to stereotypes
  - Automatic reports extracted from models to present FP size evolution to managers and board of directors

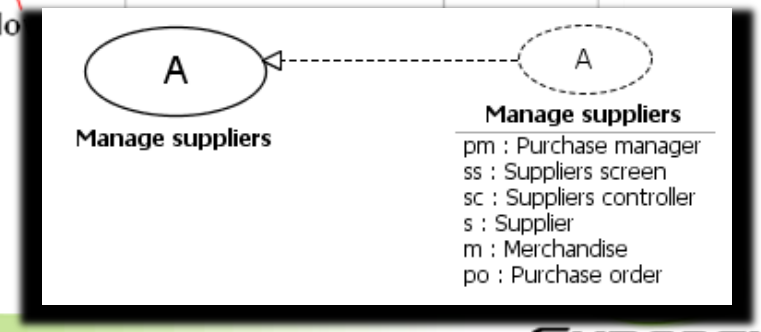
Action 3 – Count FP directly on UML Models

- There are some previous works that tried to automate FP count from UML Models [Caldiera et al., 1998] [Cantone et al., 2004] [Harput et al., 2005] [Uemura et al., 1999], but:
  - Don't guarantee accurate counts.
  - Require a detailed and very formal UML modeling, which is not possible in earlier project stages.

Action 3 – Count FP directly on UML Models

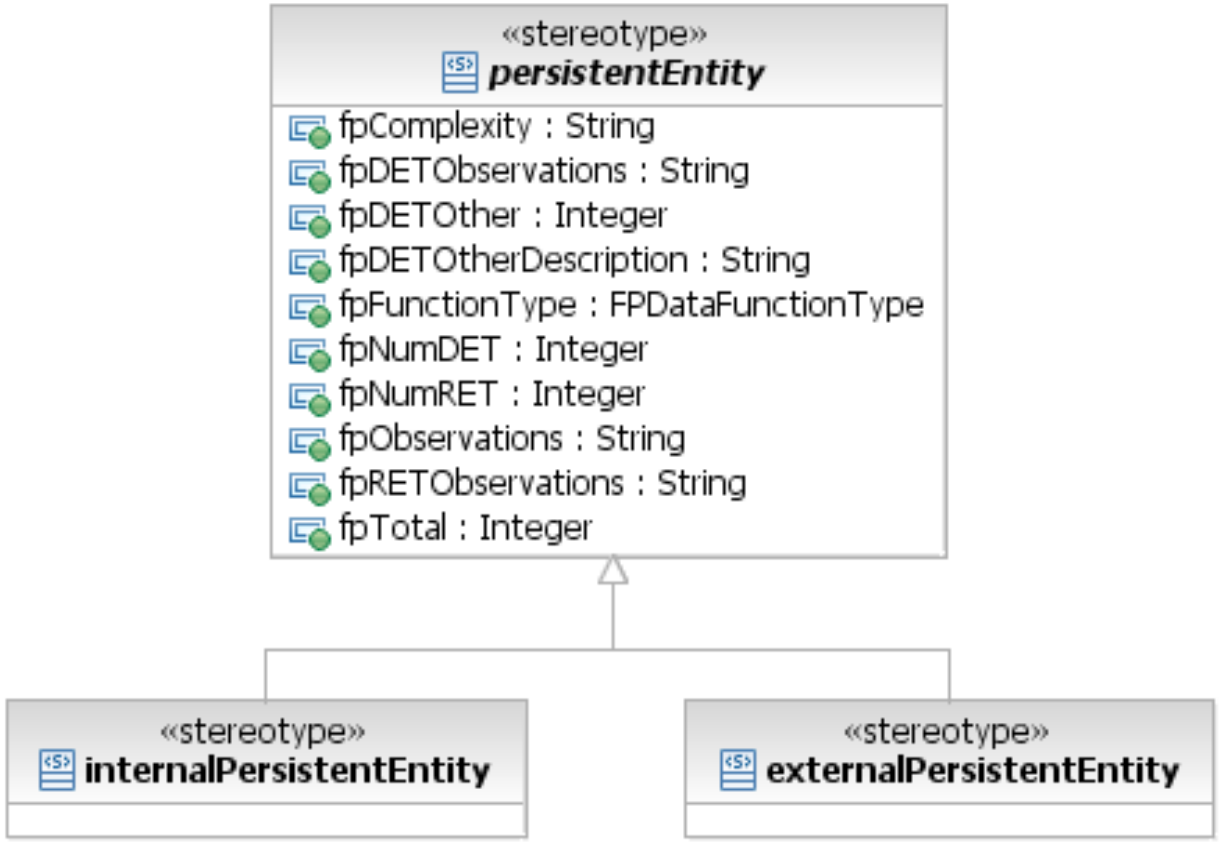






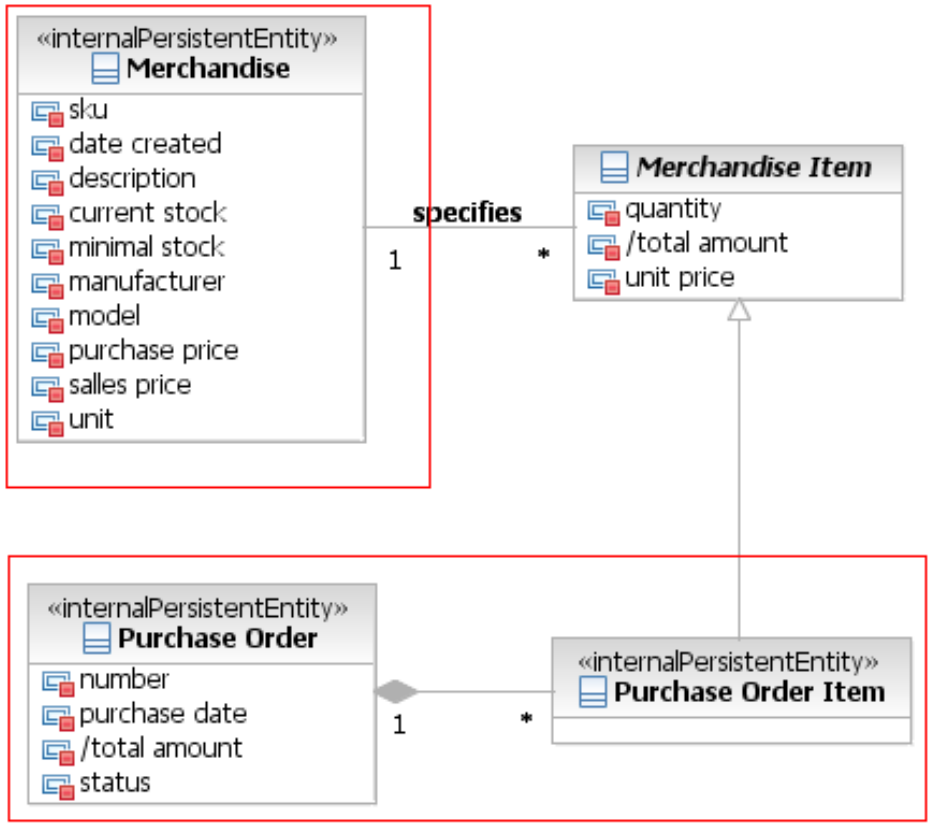
Synergia
Engenharia de Software e Sistemas

Action 3 – Count FP directly on UML Models



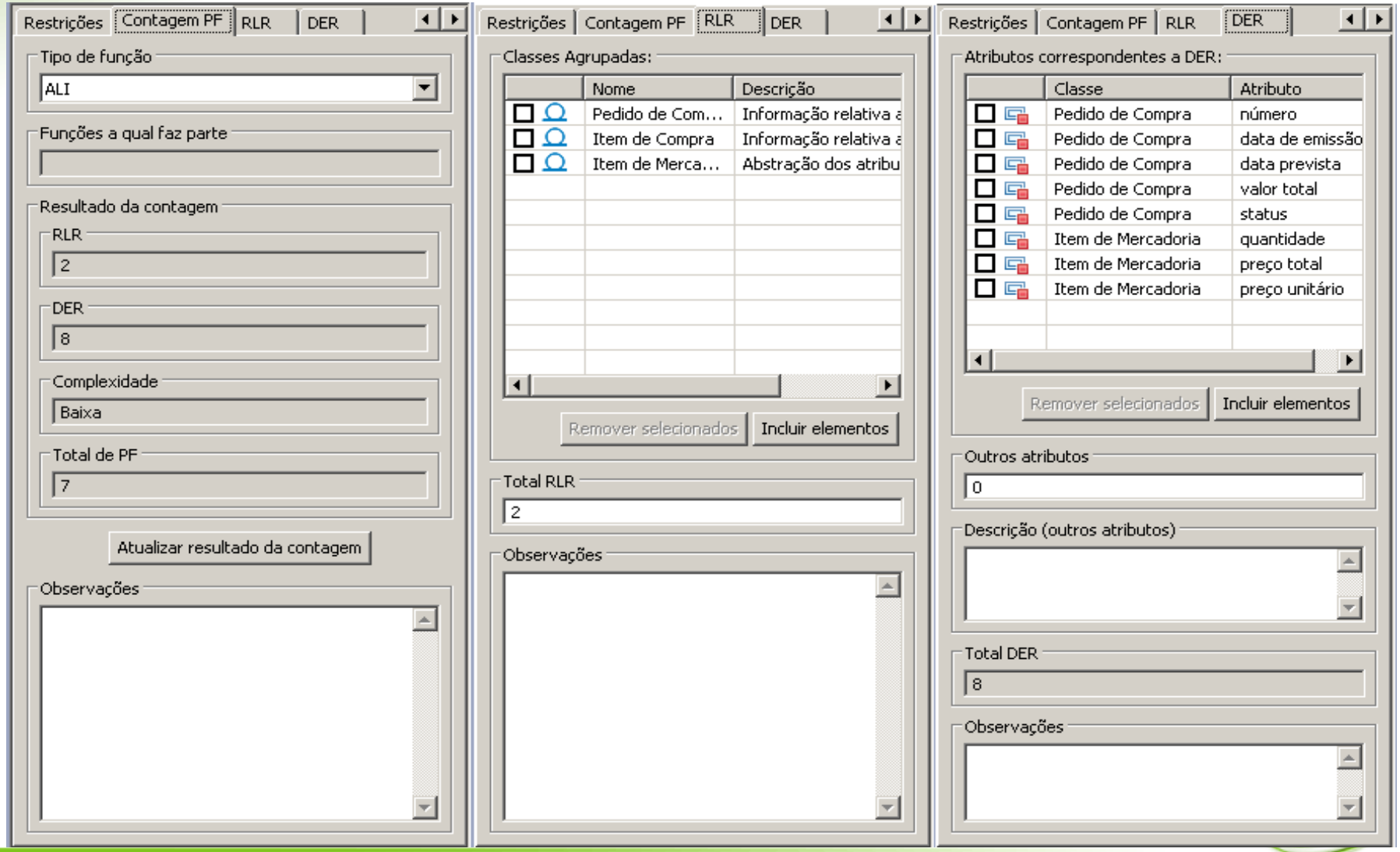
Synergia
Engenharia de Software e Sistemas

Action 3 – Count FP directly on UML Models



Synergia
Engenharia de Software e Sistemas

Action 3 – Count FP directly on UML Models



Synergia
Engenharia de Software e Sistemas

Action 3 – Count FP directly on UML Models

- **Benefits:**
  - **Functional size can be tracked during product specification.**
  - **Consistency between UML Model and FP count (previously, FP were recorded in spreadsheets).**
  - **Defect prevention in counting procedure**
  - **Major counting productivity improvement:**
    - **18.1 counted FP/h in past projects**
    - **37.6 counted FP/h in new projects**

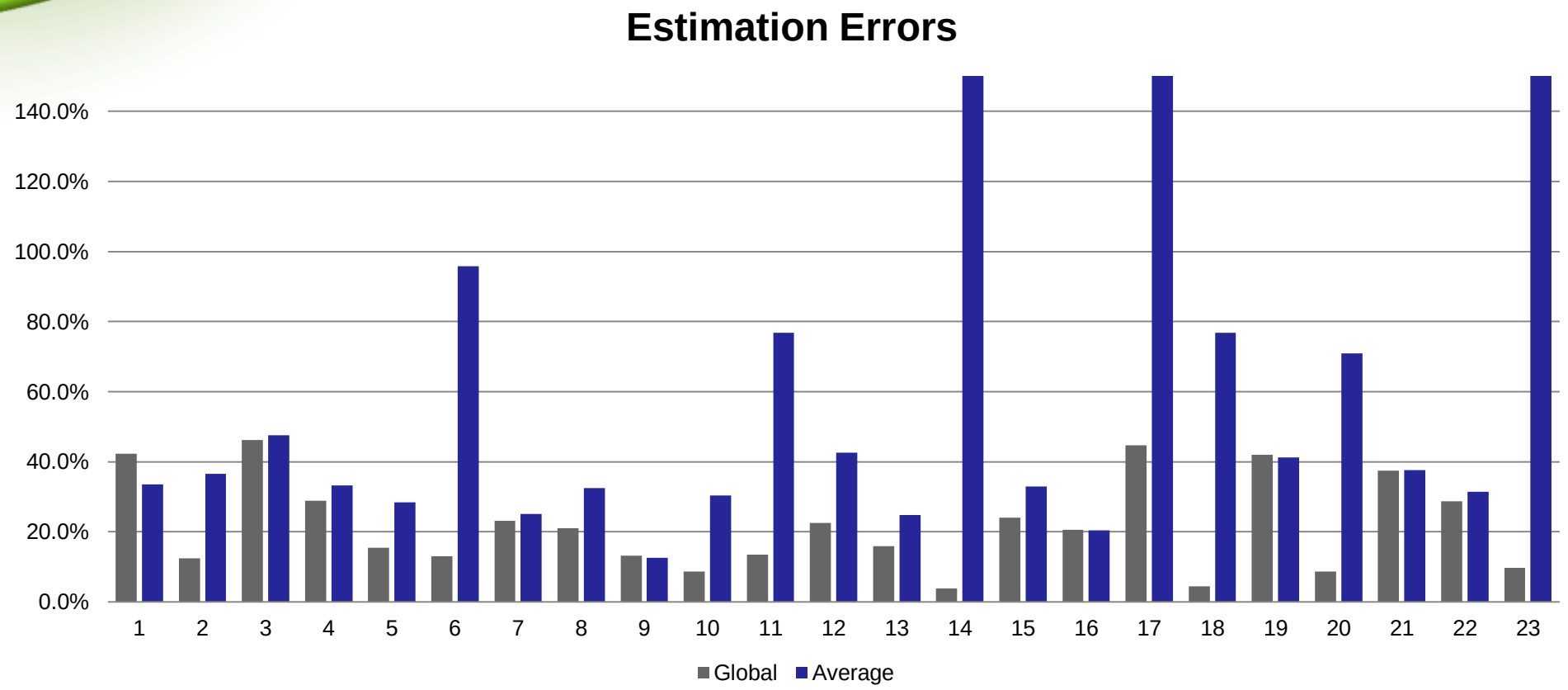
Action 4 – Detailed iteration planning

- Evaluation of two techniques proposed by McConnell [McConnell, 2006]:
  - Divide big tasks into small ones. Take advantage of the “large numbers law”
  - Individual tasks are estimated by its performer

Action 4 – Detailed iteration planning

Task	Effort		
	Estimated	Actual	MRE
Study the proposed solution design	16	4,92	225,2%
Code entity layer and business rules	30	7,98	275,9%
Code controlers	20	23	13,0%
Code data objects for test	30	20,29	47,9%
Code boundaries (screens and reports)	28	83,98	66,7%
Integrate use case to product	8	8,69	7,9%
Perform manual tests from specification	8	12,11	33,9%
Total	**140**	**160,97**	**13,0%**
		MMRE	**95,8%**

Action 4 – Detailed iteration planning



MMRE = 21,8%. **Previous: 52,8%.**

PRED(0,25) = 69,6% . **Previous: 36,8%**

Synergia
Engenharia de Software e Sistemas

Action 5 – Central project data repository

- Challenges:
  - Integrate several tools and databases into a central repository:
    - Redundant data without standardization
  - There is no tool in the market which adapts to organization's tools and databases without large integration effort [Auer *et al.*, 2003].
  - Flexible database models for software project data [Olsina *et al.*, 2002] [Harrison, 2004] that support any process are very hard to understand and to extract data

Action 5 – Central project data repository

- **Solution: Development of a Data Warehouse (DW).**
- **Kimball [Kimball & Ross, 2002] *versus* Inmon [Inmon, 2005].**
- **Information need was detailed with Synergia staff using PSM [McGarry *et al.*, 2001].**
  - **Several measures and indicators are already collected at Synergia.**

Action 5 – Central project data repository

Disciplina	Estado: Geral	10	20	30	40	50	60	70	80	90	100	Total geral
Desenho & Impl.	3,780%	0,000%	0,000%	0,000%	5,327%	0,000%	26,469%	0,000%	13,247%	0,000%	0,000%	48,822%
Eng. Processos	0,014%	0,000%	0,000%	0,000%	0,000%	0,000%	0,000%	0,000%	0,000%	0,000%	0,000%	0,014%
Gestão de Alterações	0,294%	0,000%	0,000%	0,000%	0,000%	0,000%	0,000%	0,000%	0,000%	0,000%	0,000%	0,294%
Gestão de projeto	10,936%	0,000%	0,000%	0,000%	0,000%	0,000%	0,000%	0,000%	0,000%	0,000%	0,000%	10,936%
Qualidade	0,000%	0,000%	0,000%	0,000%	0,000%	0,000%						
Req. & Análise	0,526%	0,759%	1,963%	0,024%	1,012%	0,000%						
Testes	5,584%	0,000%	0,000%	0,000%	0,742%	5,353%						
Treinamento	9,243%	0,000%	0,000%	0,000%	0,000%	0,000%						
Usabilidade	2,025%	0,273%	0,236%	0,000%	0,968%	0,000%						
Total geral	**32,401%**	**1,032%**	**2,199%**	**0,024%**	**8,048%**	**5,353%**						

Disciplina	Retrabalho: Não	Sim
Desenho & Impl.	63,540%	36,460%
Eng. Processos	100,000%	0,000%
Gestão de Alterações	92,848%	7,152%
Gestão de projeto	98,309%	1,691%
Qualidade	58,605%	41,395%
Req. & Análise	82,741%	17,259%
Testes	79,054%	20,946%
Treinamento	100,000%	0,000%
Usabilidade	82,465%	17,535%
Total geral	**75,893%**	**24,107%**

Conclusions

- Performing an SPI program using a defined process (ProMOTe) eases the hard work of changing organizational process and culture

- During SPI, several issues related to estimation are handled

Conclusions

- Some results achieved in SPI are already tested and incorporated into Synergia's processes:
  - COCOMO
  - FP count over UML models
  - Detailed iteration planning using performer estimative
  - Data warehouse (has been continuously improved)

Questions

Contact:

Vitor Alcântara Batista

Process Engineering Manager

vitor@dcc.ufmg.br

www.synergia.dcc.ufmg.br

References

- [Auer et al., 2003] Auer, M., Graser, B., & Biffl, S. (2003). A survey on the fitness of commercial software metric tools for service in heterogeneous environments: common pitfalls. In Software Metrics Symposium, 2003. Proceedings. Ninth International (pp. 144-152).
- [Boehm et al., 2000] Boehm, B. W., Abts, C., Brown, A. W., Chulani, S., Clark, B. K., Horowitz, E., et al. (2000). Software Cost Estimation with COCOMO II. Prentice Hall PTR.
- [Caldiera et al., 1998] Caldiera, G., Antoniol, G., Fiutem, R., & Lokan, C. (1998). Definition and Experimental Evaluation of Function Points for Object Oriented Systems. Procs. IEEE METRICS-98.
- [Cantone et al., 2004] Cantone, G., Pace, D., & Calavaro, G. (n.d.). Applying function point to unified modeling language: Conversion model and pilot study. Proceedings of the 10th International Symposium on Software Metrics (METRICS'04), 280–291.
- [Harput et al., 2005] Harput, V., Kaindl, H., & Kramer, S. (2005). Extending function point analysis to object-oriented requirements specifications. In Software Metrics, 2005. 11th IEEE International Symposium (pp. 10 pp.-39).

References

- [Harrison, 2004] Harrison, W. (2004). A flexible method for maintaining software metrics data: a universal metrics repository. Journal of Systems and Software, 72, 225-234.
- [Inmon, 2005] Inmon, W. H. (2005). Building the Data Warehouse. Wiley Publishing, Inc.
- [Kimball & Ross, 2002] Kimball, R., & Ross, M. (2002). The Data Warehouse Toolkit. Wiley Computer Publishing.
- [Laird, 2006] Laird, L. M. (2006). The Limitations of Estimation. IT Professional, 8, 40-45.
- [McConnell, 2006] McConnell, S. (2006). Software Estimation, Demystifying the Black Art. Microsoft Press.
- [McGarry et al., 2001] McGarry, J., Card, D., Jones, C., Layman, B., Clark, E., Dean, J., et al. (2001). Practical Software Measurement: Objective Information for Decision Makers. Addison-Wesley Professional.

References

- [Nan & Harter, 2009] Nan, N., & Harter, D. E. (2009). Impact of Budget and Schedule Pressure on Software Development Cycle Time and Effort. IEEE Transactions on Software Engineering, 35(5), 624 - 637.
- [Olsina et al., 2002] Olsina, L., Lafuente, G., & Pastor, O. (2002). Towards a reusable repository for Web metrics. Journal of Web Engineering, 1, 61-73.
- [Uemura et al., 1999] Uemura, T., Kusumoto, S., & Inoue, K. (1999). Function point measurement tool for UML design specification. In Software Metrics Symposium, 1999. Proceedings. Sixth International (pp. 62-69).
- [Yang et al., 2008] Yang, D., Wang, Q., Li, M., Yang, Y., Ye, K., Du, J., et al. (2008). A survey on software cost estimation in the chinese software industry. In ESEM '08: Proceedings of the Second ACM-IEEE international symposium on Empirical software engineering and measurement (pp. 253-262). New York, NY, USA.