

# Continuing Effort Estimation for Evolution - A Case Study

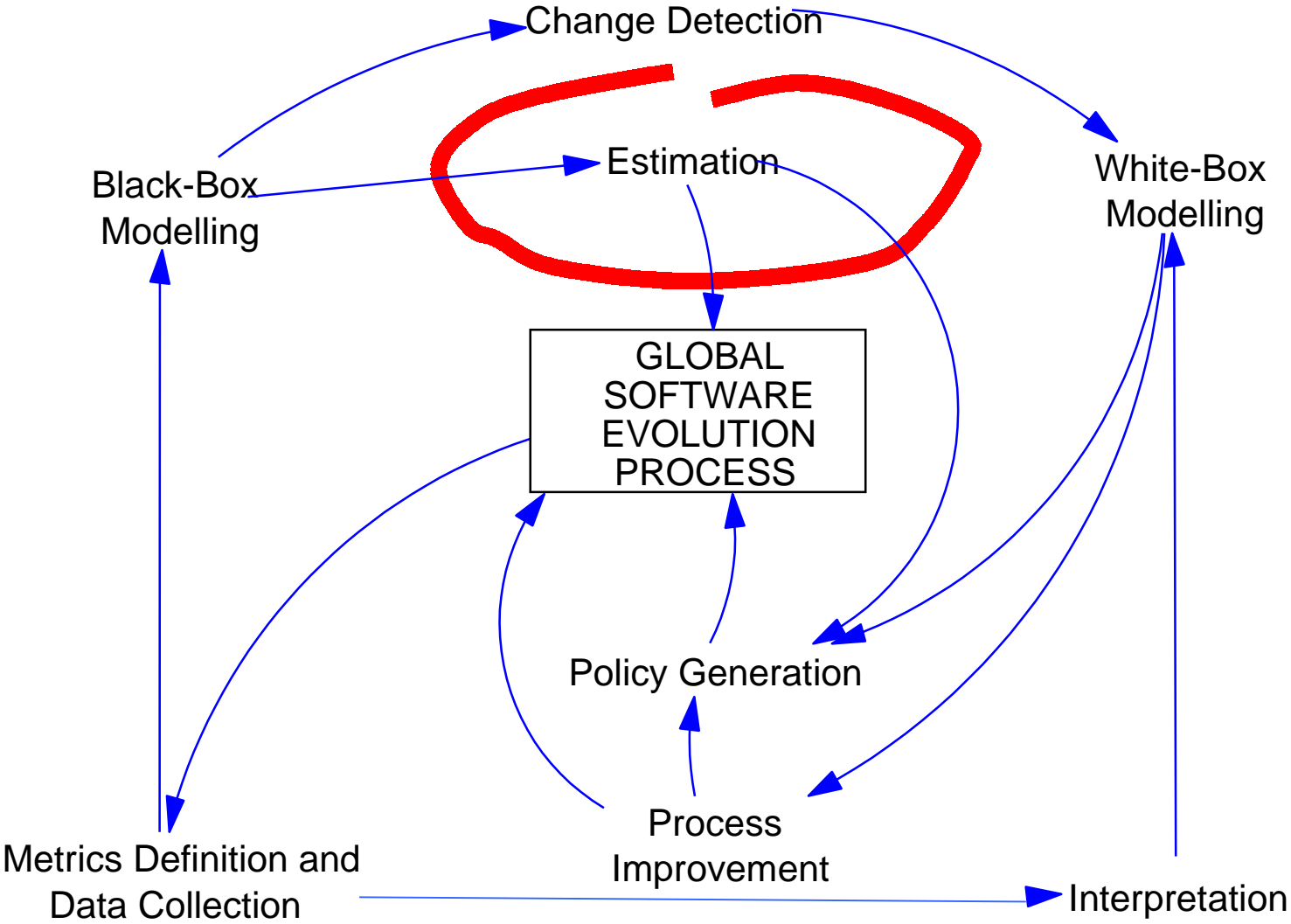
University of Sannio  
3 May 2001

**Juan F Ramil**

Department of Computing  
Imperial College  
180 Queen's Gate  
London SW7 2BZ  
tel. +44 (0) 20 7594 8216  
fax. +44 (0) 20 7594 8215

ramil@doc.ic.ac.uk  
<http://www.doc.ic.ac.uk/~ramil>

# Case Study - Resource Estimation



## Effort estimation models

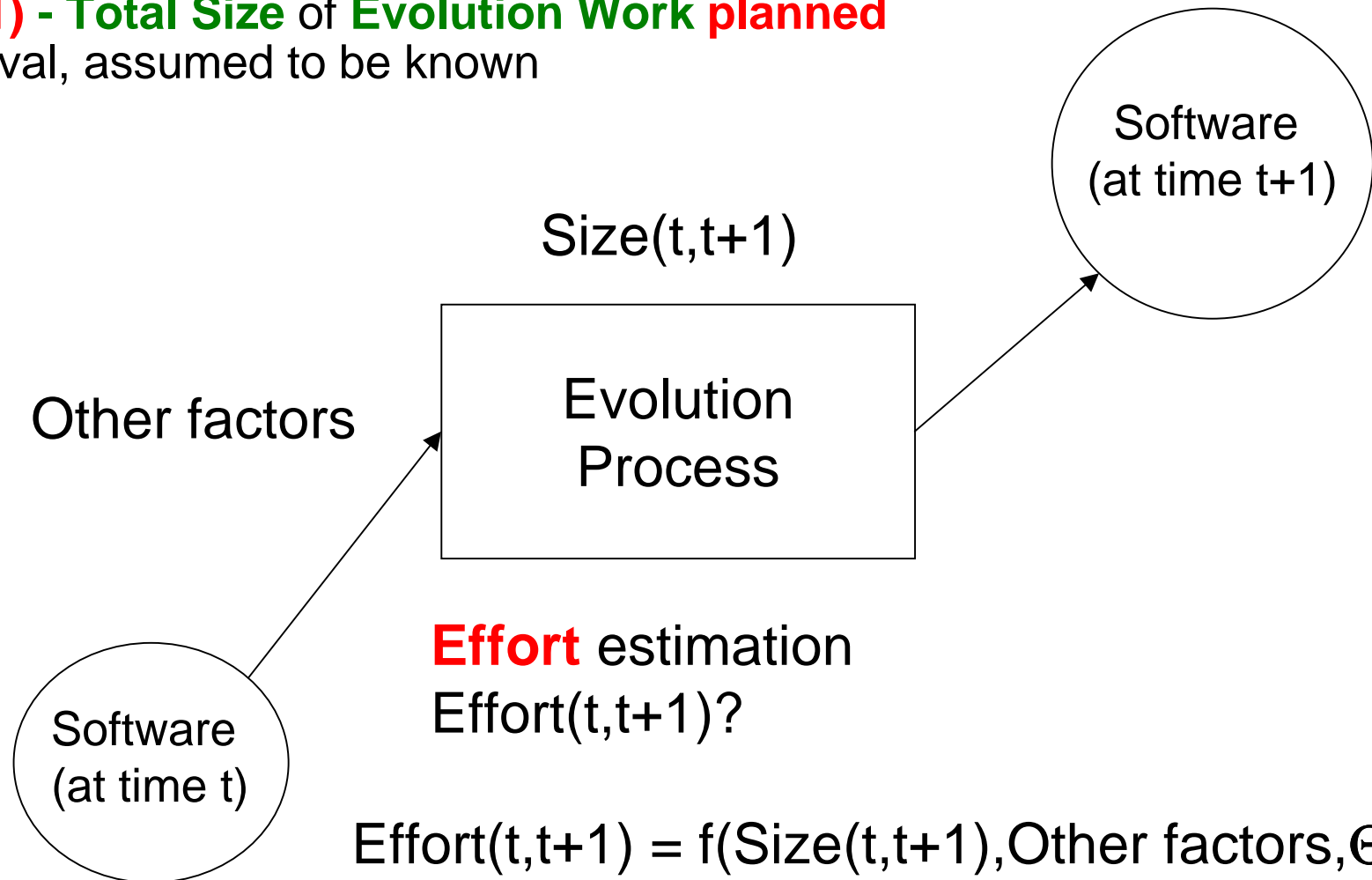
- useful on their own
- provide “surrogate” indicators of effort for system dynamics modelling over segments of evolution time

## Indicators of Evolution Activity Used in Study

<b>Indicator Based on</b>	<b>Abbreviation (Full name)</b>	<b>Description</b>
Modules	<i>ModifHandlings(t)</i> (Modification Handlings)	Symbol '#' to be read as 'count over interval $t$ to $t+1$ ' # of changes to modules as reflected by number of change log modification entries, referred to as <i>handlings</i>
Modules	<i>ModulesChanged(t)</i> (Modules Changed)	# of modules modified
Modules	<i>ModulesCreated(t)</i> (Modules Created)	# of modules added to the system
Modules	<i>TotalHandlings(t)</i> (Total Number of Handlings)	# of total change log entries, that is, including both creation entries and modification entries
Modules	<i>ModulesHandled(t)</i> (Modules Handled)	# of modules, either added to the system or modified, or both (if both, module is counted once)
Subsystems and Modules	<i>SubsysChanged(t)</i> (Subsystems Changed)	# of subsystems that underwent modifications to their modules
Subsystems and Modules	<i>SubsysHandled(t)</i> (Subsystems with Modules Handled)	# of subsystems that underwent either additions or changes to their modules, or both (if both, subsyst. is counted once)
Subsystems and Modules	<i>SubsysInclCreations(t)</i> (Subsystems that include Modules Created)	# of subsystems which underwent module additions

## View for Real-time Aggregated Data

**Size(t,t+1)** - Total Size of **Evolution Work planned** over interval, assumed to be known

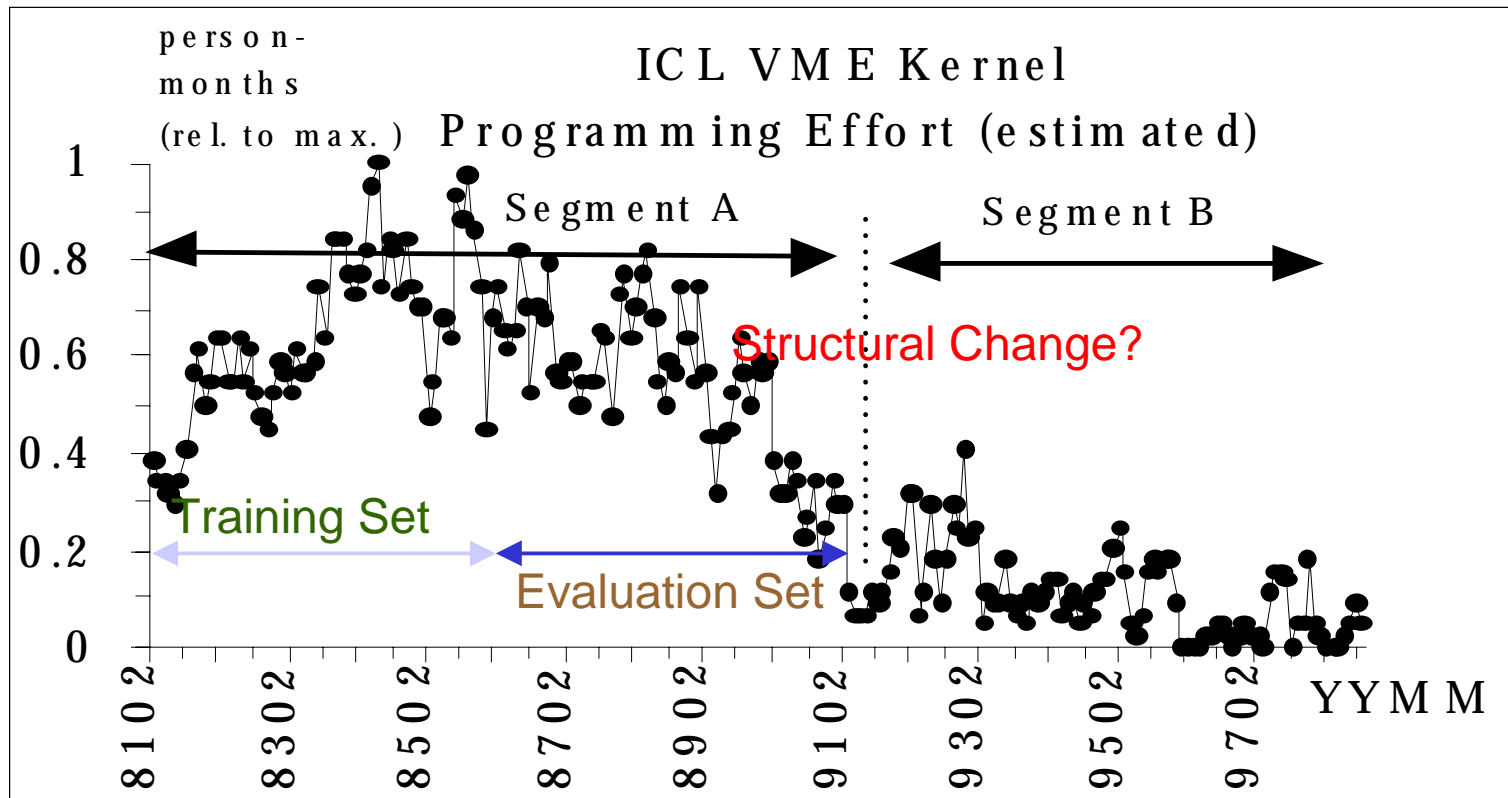


## Case Study

- **ICL VME** Mainframe Operating System -
  - Two parts: **Kernel** and *Director*
  - Data made available for Kernel only - low level, hardware related functionality
- **Continuing** evolution over more than 20 years - system **adapted** and **enhanced** under technological and market pressures, remarkable personnel stability
- Implemented in **SDL** - S3 as underlying language, similar to **Algol68**
- Some 60,000 **change entries** from 1981 to 1998 - semiautomatic collection as part of CADES
- An **entry** is generated each time a developer submits an artefact to common workspace - code check in
- Recording of such **submissions** made compulsory after 1986 or so
- *Perl* scripts developed for data extractions
  - **Effort applied, Size** - focus on comparing several metrics
  - **Other factors** - left for the future
  - Some 200 observations obtained - one per month

## Case Study

- **Effort** applied
  - Effort applied derived from initials of developers in **change history records**
- Visual inspection of trends suggest 2 distinct segments - 1981 to 1990 and 1991 - 1998
- Case study focused on **first segment** - 5 years training, 5 years for model assessment



## Models

Model	Expression
M1 Base- line 1	Average effort over training set
M2 Base- line 2	$ModulesHandled(t) \times [Avg. \text{ effort per module handled over training set}]$
M3	$A.ModulesHandled(t) + B$
M4	$A.SubsysHandled(t) + B$
M5	$A.TotalHandlings(t) + B$
M6	$A_1.ModulesCreated(t) + A_2.(ModulesChanged(t) - ModulesCreated(t)) + B$
M7	$A_1.ModulesCreated(t) + A_2.(ModifHandlings(t) - ModulesCreated(t)) + B$
M8	$A_1.SubsysInclCreations(t) + A_2.SubsysChanged(t) + B$

## Case Study

Model assessment over evaluation set based on following indicators:

- **MMRE** – Mean Magnitude of Relative Error, defined as *average* over test set of  $(|Predicted Effort - Actual Effort| / Actual Effort)$
- **MdMRE** – Median Magnitude of Relative Error, defined as *median* over test set of  $(|Predicted Effort - Actual Effort| / Actual Effort)$
- **PRED(10)** – Number of observations in test data set for which MMRE is equal or lower than 10 %
- **PRED(25)** – Number of observations in test data set for which MMRE was equal or lower than 25 %

## Predictive Power

	Model	MMRE In per cent.	MdMRE In per cent.	PRED (10)	PRED (25)
Baseline 1	M1	39.7	17.9	18	39
Baseline 2	M2	35.5	36.7	5	14
Univ Reg.	M3	19.6	14.9	18	<b>47</b>
"	M4	32.8	18.8	10	39
"	M5	21.6	18.7	17	42
Multiv Reg	M6	19.6	16.3	12	<b>47</b>
"	M7	20.0	15.8	21	45
"	M8	<b>19.3</b>	<b>14.0</b>	<b>22</b>	44

## Summary

- Models fitted and tested to historical data over **one segment** of 10 years of continuing evolution
- 5 years of data to **calibrate** models, 5 years to **assess** predictive power
- Predictive **accuracy** of **20 percent** of actual values for 5 of 6 models
- **Subsystems vs modules**
  - **coarse** granularity measures performed similarly to **finer** ones
  - **hypothesis**: higher level metrics will perform better than LOC-based
- Results consistent with fourth law of software evolution - organisational stability
- Segments suggest relation with Rajlich and Bennett's view
- Further **case studies** and refinement of a general procedure
- Current view of such procedure follows

## A Procedure for Building Effort Estimation Models

1. Set **goals** and **extract** metric data from historical records,
  - starting with measures of effort applied and evolution activity rate
  - 'Other factors' may be considered during iterations of this procedure
2. **Compile** historical records
  - as derived from, for example, configuration management databases, change-log records
3. **Plot** data, detect segments of regular activity rate, detect and appropriately treat outliers
4. **Identify** model structure - linear, log-linear
5. **Obtain** model parameters for each of the relevant segments
6. Assess the **predictive power** of the model
  - using preferably a data set different than the one used to calculate model parameters

## A Procedure (cont.)

7. If the model is precise enough for its purpose, **accept** it as an estimation source
8. If models are not reasonably accurate,
  - **refine** metrics list and the models until an acceptable level is reached or no further significant improvement can be economically be achieved
9. Periodically, **check** whether the model **remains accurate**.
  - If there has not been significant **structural** process change, re-calibrate existing model. Finally, if necessary, explore refinement of the existing model or a different model structure
10. Periodically, **check for** large changes in activity rate, effort applied, and any **structural changes**. Update or refine models once sufficient data on the new segment has been collected