Proposals for project collection and classification from the analysis of the ISBSG Benchmark 8

GUFPI-ISMA SBC (Software Benchmarking Committee):
Domenico Natale, Luca Santillo, Italo Della Noce, Monica Lelli, Stefania Lombardi, Guido Moretto, Simonetta Ortona

Abstract:

This work presents statistical analysis scenarios of a sample of software development and enhancement projects contained in the Benchmark 8 by ISBSG (International Software Benchmarking Standards Group, 2003). The research – started in the second half of 2003 – is an on-going, incremental process based on the voluntary participation of the members of the SBC (Software Benchmarking Committee) of GUFPI-ISMA (Italian Software Metrics Association).

This work is focused on the distribution of development and enhancement projects in the ISBSG sample with regard to: sizing method, project size, completion date, productivity, and primary programming language.

Some considerations and hints arise from this research to improve the data collection process (by ISBSG as well as by any private organization within its own benchmarking database). Categorization criteria and taxonomies are identified and suggested for standardized use in order to clarify the definition and the comparison of future software projects.

Keywords

Benchmarking, software project, size, productivity, programming language

1 Introduction

GUFPI-ISMA (Gruppo Utenti Function Point Italia - Italian Software Metrics Association) is a non-profit organization, whose mission is to promote and encourage the use of software measurement methods in Italy. Through GUFPI-ISMA's activities and initiatives, leading industry experts, practitioners, and technology vendors meet regularly to share experiences, standards and visions on the software measurement topics. Several committees and workgroups in the association support the advancement of software measurement discipline and are involved in activities providing value for GUFPI-ISMA members, as well as outside the association. The GUFPI-ISMA Software Benchmarking Committee, under the guidance of Domenico Natale and Luca Santillo, is aimed to study methods, techniques and approaches to analyse and compare software performances, with special attention to productivity and cost. By means of the SBC, GUFPI-ISMA is supporting the collection and maintenance of benchmarking databases of software project information and the publication of reports and electronic media containing information about such databases and how to exploit them.

The International Software Benchmarking Standards Group is a non-profit organization, whose mission is “to help improve the management of IT resources through improved project estimation, productivity, risk analysis and benchmarking”. ISBSG maintains two repositories of software metrics: the Software Development and Enhancement repository, also known as Benchmark, and the new Software Maintenance and Support repository. Both repositories are made of anonymous data contribution by software developers from any country. Project data are sent to ISBSG by means of a questionnaire, whose format is customised in the project size section, based on the specific measurement method (COSMIC-FFP, IFPUG & NESMA, MkII, Other). The up-to-date development and enhancement collection is the Benchmark 8 (February 2003) – a collection of 2,027 software projects, described by over 50 variables.

GUFPI-ISMA is a member of ISBSG and supports its activities. In the second half of 2003, the GUFPI-ISMA SBC started a series of analysis on the ISBSG Benchmark 8. Many analyses have been already performed and published by the ISBSG itself or other authors, on previous versions and/or the current one – the SBC’s aim is to diffuse, extend, validate and enhance such kind of analysis.

In the current phase, a subset of main variables was extracted and analysed; the chosen variables are: *measurement method*, *project type*, *project size*, *development platform*, *completion date*, *work effort*, *project delivery rate*, and *programming language*. Further analyses will take into account more variables and eventually more possible correlations and regression will be investigated and reported in future publications by the SBC. Although this is just the first step of the SBC’s analysis plan, some suggestions already arise in order to improve and enhance the collection and presentation of the benchmarking data, in order to provide more effective and complete analysis results, in terms of both quality and quantity.

GUFPI-ISMA SBC have analyzed the data from ISBSG database with the intent of better understand the meaning, usefulness and consistency of it. Therefore, the published results are not to be considered as a valid reference for any possible official, commercial or legal utilization. Neither GUFPI-ISMA, nor the authors can be hold responsible for errors or damages coming form external utilization of the analysis results.

2 Demographic overview

This section contains a brief overview of the overall Benchmark 8 database. Subsequent analyses were performed on specific data subsets, selected by means of filters to represent significant information. Throughout the paper, graphs and tables are presented – with the number of projects involved – in order to report some key statistics of selected variables. Variables not explicitly involved in the analyses, are not described in further details (please, refer to the ISBSG documentation for details).

The following header is common to many of the tables presented:

N	Min	P10	P25	Median	P75	P90	Max	Mean	Std Dev

where:

- N is the number of cases or data instances in the sample or sub-sample.
- Min is the minimum value found in the sample or sub-sample.
- P10 is the 10th percentile (it is that value which is greater than the values of ten percent of the members of the sample or sub-sample).
- P25 (also known as Q1) is the 25th percentile – or first quartile.
- Median (also known as P50) is the middle value, half the values in the data sample or sub-sample are below this value, while the other half have values which are greater.
- P75 (also known as Q3) is the 75th percentile – or third quartile.
- P90 is the 90th percentile.
- Max is the maximum value found in the sample or sub-sample.
- Mean is the arithmetic mean or average.
- Std Dev is the standard deviation.

Note that the median is a more useful measure, with respect to the mean, when the data contain outliers or when they are strongly skewed, as is often the case here. Percent values, as for example the percent of cases included in a specific sub-sample with respect to the higher sample, are also expressed. The maximum value of N is 2,027, since this is the total number of project instances in the Benchmark 8 – but this amount is rarely reached, since the database contains values which are void, unknown , or unclear, under several variables.

All the analyses have been performed with the aid of an electronic spreadsheet; median, mean, percentile and standard deviation values have all been obtained by means of the functions provided by the spreadsheet application.

Table 1 (next pages) reports an overview of the ISBSG Benchmark 8 database, including composition and consistency of each variable. Note that the “value range” column reports the number of different value instances, not the list of such values.

Variable	ISBSG Name	N	%	Value Type	Value Range	Multiple values	Calc
PRJ_ID	Project ID	2,027	100.0%	Code	-		
DQR	Data Quality Rating	2,027	100.0%	Ord.	4		
MM	Derived Count Approach	2,024	99.9%	Text	14		
FP	Function Points	2,011	99.2%	Num.	-		
VAF	Value Adjustment Factor	1,382	68.2%	Num.	-		
FP_STD_PRIMARY	FP Standard	1,938	95.6%	Text	25		
COUNT_TECH	Counting Technique	1,436	70.8%	Text	10		
RTAB_APPR	Reference Table Approach	1,263	62.3%	Text	34		
WE_TOT	Summary Work Effort	2,025	99.9%	Num.	-		
WE_METH	Recording Method	1,946	96.0%	Text	5		
RL	Resource Level	2,027	100.0%	Ord.	4		
MTS	Max Team Size	1,015	50.1%	Num.	-		
PRJ_TYPE	Development Type	2,027	100.0%	Text	5		
PLATFORM	Development Platform	1,418	70.0%	Text	3		
-	Architecture	-	-	?	?		
GL	Language Type	1,636	80.7%	Ord.	5		
LANG	Primary Programming Language	1,691	83.4%	Text	122	Yes	
DBMS	DBMS Used	1,473	72.7%	Y/N	1		
UCASE	Upper CASE Used	1,073	52.9%	Y/N	3		
LCASE_WCGEN	Lower CASE Used (with code generator)	1,071	52.8%	Y/N	3		
LCASE	Lower CASE Used (without code generator)	1,019	50.3%	Y/N	3		
ICASE	Integrated CASE Used	1,031	50.9%	Y/N	3		
METH	Used Methodology	1,780	8.8%	Y/N	3		
METH_HOW	How Methodology Acquired	1,035	51.1%	Text	3	Yes	
DT	Development Techniques	1,025	50.6%	Text	227	Yes	
PRJ_TIME	Project Elapsed Time	1,639	80.9%	Num.	-		
PRJ_ITIME	Project Inactive Time	701	34.6%	Num.	-		
DATE_IMPL	Implementation Date	1,802	88.9%	Date	-		
DEF_EXTR	Extreme Defects	250	12.3%	Num.	-		
DEF_MAJ	Major Defects	398	19.6%	Num.	-		
DEF_MIN	Minor Defects	326	16.1%	Num.	-		
USER_UNITS	User Base – Business Units	588	29.0%	Num.	-		
USER_LOCS	User Base – Locations	607	29.9%	Num.	-		
USER_CONC	User Base – Concurrent Users	589	29.1%	Num.	-		

(cont.)

Variable	ISBSG Name	N	%	Value Type	Value Range	Multiple values	Calc
OT	Organisation Type	1,502	74.1%	Text	92	Yes	
BAT	Business Area Type	1,157	57.1%	Text	98	Yes	
AT	Application Type	1,471	72.6%	Text	114	Yes	
PACKAGE	Package Customisation	1,322	65.2%	Y/N	3		
PACKAGE_CUSTDEG	Degree of Customisation	87	4.3%	Text	18		
PRJ_SCOPE	Project Scope	1,275	62.9%	Text	26	Yes	
WE1P	WE Plan	957	47.2%	Num.	-		
WE2S	WE Spec	1,180	58.2%	Num.	-		
WE4B	WE Build	1,291	63.7%	Num.	-		
WE5T	WE Test	1,245	61.4%	Num.	-		
WE6I	WE Impl	846	41.7%	Num.	-		
WE_PRJ_NONPRJ	Ratio Prj. Work Effort to Non-Prj. Activity	449	22.2%	Text	55		
WE_PERC_NONPRJ	(transformed from previous variable)		idem	Num.	-		Yes
WE_UNCOLL	Percentage of Uncollected Work Effort	191	9.4%	Text	16		
FP_EI	Input count	2,027	100.0%	Num.	-		
FP_EO	Output count	2,027	100.0%	Num.	-		
FP_EQ	Enquiry count	2,027	100.0%	Num.	-		
FP_ILF	File count	2,027	100.0%	Num.	-		
FP_EIF	Interface count	2,027	100.0%	Num.	-		
FP_ADD	Added	2,027	100.0%	Num.	-		
FP_CHG	Changed	2,027	100.0%	Num.	-		
FP_DEL	Deleted	2,027	100.0%	Num.	-		
SLOC	(Source) Lines of Code	287	14.2%	Num.	-		
DEF_TOT	Total Defects Delivered	466	23.0%	Num.	-		Yes
WE_UNPH	Work Effort Unphased	2,027	100.0%	Num.	-		Yes
WE_NORM	Normalised Work Effort	2,024	99.9%	Num.	-		Yes
UFP	Unadjusted Function Points	2,027	100.0%	Num.	-		Part.
UFP_RAT	Unadjusted Function Point Rating	2,027	100.0%	Ord.	4		Yes
FP_STD_ALL	FP Standards All	1,946	96.0%	Text	67	Yes	?
PDR_NORM	Normalised Project Delivery Rate	1,569	77.4%	Num.	-		Yes
PDR	Project Delivery Rate	1,569	77.4%	Num.	-		Yes
APDR_NORM	Normalised Project Delivery Rate (adj'd)	2,011	99.2%	Num.	-		Yes
APDR_REP	Reported Project Delivery Rate (adj'd)	2,011	99.2%	Num.	-		Yes

Table 1: Characteristics of collected variables in the overall sample; critical aspects are highlighted in grey background – see section 2.2.

2.1 Data origin

The source of the projects is not reported in the database, due to the anonymity reasons. Quoting form ISBSG documentation, we find that the projects "come from 20 different countries". Major contributors are: Australia, Japan, the United States, the Netherlands, Canada, and United Kingdom; among smaller contributors: India, France, Brazil, Italy, and others.

2.2 Analysis of variables collection

Each of the 66 variables - collected by means of questionnaires by ISBSG – have been analysed. Here's a list of comments on observed data:

- only 15 variables out of 66 contain values for more than 90% of the sample (note that variables from "FP_EI" to "FP_DEL" and calculated variable "UFP" have many instances filled with zero values);
- "Architecture", cited by ISBSG in documentation describing the benchmark, is not present in the database spreadsheet;
- "Language Type" (i.e. programming language generation) and "Implementation Date" contain ambiguous values (e.g. projects with same primary programming language classified as 3GL as well as 4GL; periods of time instead of exact dates);
- 8 variables (Table 1, "Multiple values = Yes") report multiple values, that is the range of their values is complicated by a combination of a smaller set of elementary values (e.g. the Primary Programming Language field reporting more than one language, or the Project Scope field listing different combinations of elementary project phases);
- variables of textual type often show several typing discrepancies, as expressions which are similar but not identical (e.g. "COBOL 2", "COBOL II" and "COBOL V2", or "COBOL for MVS" and "MVS COBOL", or "VB", "VBASIC" and "VISUAL BASIC", or "IFPUG 4" versus "IFPUG 4.0" or "IFPUG 4.1", and so on);
- some "Y/N/Don't Know" type variables contain many void instances (and no "No" values, in some case), which could be interpreted as both "No" and "Don't Know" values;
- some variables carrying quantitative information, as the "Degree of Customisation", are expressed in a free textual form that makes very difficult to use them without the risk of misinterpretation.

Along with the analyses reported in this paper, the SBC decided to start the collection of a Glossary of benchmarking terms, to clarify the meaning and the interpretation of each variable.

Next, some data manipulation was performed to resolve some of the critical aspects above:

- *dichotomization*: variables which show combination of multiple values have been dichotomized – or "binarised" (with the exception of the Primary Programming Language); for example, the Project Scope field, showing almost every possible combination of "Planning", "Specification", "Build", "Test", and "Implement" phases (value range = 26), was replaced by five variables of Y/void type (one per each phase), and the same has been performed for:
  - "FP Standards All" (22 Y/N variables vs. a value range of 67),
  - "Development Techniques" (20 Y/N variables vs. a range of 227),
  - "Organization Type" (37 Y/N variables vs. a range of 92),
  - "Business Area Type" (18 Y/N variables vs. a range of 98),
  - "Application Type" (24 Y/N variables vs. a range of 114).
- *nomenclature*: subject to prior discussion and approval for each case, expressions that were similar and interpreted as equivalent were rendered with identical terms (so, "COBOL 2", "COBOL II" and "COBOL V2" were all rendered as "CONBOL 2", and so on – also, some self-evident repetition of values were eliminated, as "Management Information System; Management Information System" in the "Application Type" field, and so on).

Other suggestions arising from this preliminary analysis regard:

- *taxonomy*: define *closed ranges* in case of several possible values, particularly for textual type variables (leaving the possibility of "*other*" assignment, with a *separate note* to describe values which are not included in the prefixed range), in order to reduce the presence of spurious or unclear textual values;
- *completeness*: insist on the *mandatory filling* of every and each project attribute, when submitting the questionnaires, in order to reduce the amount of void values, i.e. the percentage of incomplete records.

2.3 Overall values distributions for selected variables

The selected variables for the analyses by SBC are:

- size (UFP)
- work effort (WE_TOT)
- project delivery rate (PDR)
- platform (PLATFORM)
- primary programming language (LANG), and
- implementation date (IMPL_DATE)

Simple distribution analyses are reported for selected variables over the whole ISBSG sample. *Other* variables are used – in next section – as filtering criteria in order to obtain restricted samples for significant analyses of selected variables. Note that project size, effort and (consequently) delivery rate distributions are – possibly strongly – skewed towards low values. In the overall sample, size is mixed over different measurement methods (IFPUG, COSMIC, MkII, etc.); same implementation date values are not in term of an exact date, but rather as only year or period (e.g. “1998-2000”, interpreted as implementation year = “2000”).

UFP (unadjusted function point size – measurement unit: UFP)

N	%	Min	P10	P25	Median	P75	P90	Max	Mean	Std Dev
1,568	77.4	6.0	63.0	109.8	224.0	476.0	1,182.2	19,050.0	514.2	1,087.3

Since this initial sample mixes different measurement methods and project types and contains clear outliers (compare the 90th percentile with the maximum value), it is not significant to investigate the plot of such distribution, unless some filtering is performed.

WE_TOT (summary work effort – measurement unit: ph)

N	%	Min	P10	P25	Median	P75	P90	Max	Mean	Std Dev
2,025	99.9	5.0	419.4	888.0	2,200.0	5,307.0	13,737.6	645,694.0	6,883.4	26,160.8

As for previous size overall distribution (different measurement methods, different project types, and clear outliers), it not significant to investigate the plot of such distribution.

PDR (project delivery rate – measurement unit: ph/UFP)

N	%	Min	P10	P25	Median	P75	P90	Max	Mean	Std Dev
1,569	77.4	0.02	2.0	4.2	9.0	18.0	33.7	640.0	15.6	26.7

Same as above; the distribution shows regularity, but mixed types introduce a tail of high values; investigation requires some filtering.

PLATFORM (development platform – textual variable)

N	%	MF	%_N	MR	%_N	PC	%_N
1,418	70.0	844	59.5	252	17.8	322	22.7

Legenda: MF = Mainframe, MR = Midrange, PC = Personal Computer

Some doubts arose about the exact meaning of the three proposed classes for platform range; further definitions are needed.

LANG (primary programming language – textual variable)

N	%	Cobol*	%_N	C	%_N	VB	%_N	C++	%_N	Oracle	%_N	rest	%_N
1,691	83.4	451	26.7%	153	9.0%	115	6.8%	114	6.7%	108	6.4%	751	44.4%

Only largest frequencies are shown; “COBOL*” stands for variants of Cobol (COBOL, COBOL 2, MVS COBOL, etc.); “rest” includes approximately seventy languages (SQL, PL, , NATURAL, ACCESS, JAVA, etc.).

IMPL_DATE (implementation date – date field)

N	%	2002	%_N	2001	%_N	2000	%_N	1999	%_N	1998	%_N	rest	%_N
1,802	88.9	147	8.2%	75	4.2%	387	21.5%	282	15.6%	236	13.1%	675	37.5%

Only last five years are shown; some records report only the year, not the exact date; some values are expressed as a range (e.g. “1998-2000” was interpreted as implementation date falling in the 2000); “rest” includes dates in the range 1989-1997.

3 Subsets selection for analyses

In order to analyse the selected variables, some filtering and transformation actions had to be performed on the original ISBSG data sample, leading to two sub-samples, denoted as sample A (or “soft filter” sub-sample) and sample B (or “severe filter” sub-sample). Moreover, the “project type” attribute has been kept in the sub-samples to differentiate the analysis results by project type: development or enhancement.

3.1 Selection criteria

Table 2 (below) reports variables and corresponding criteria used for filtering. It was decided to consider two sub-samples in order to compare the effect of high reduction of residual records when applying stronger filters on both distribution and correlation analyses of the selected variables.

Note that filtering out records with PACKAGE = “Y” left records with both explicit “N” values and void values, so that a remaining impact of undocumented package customisation for some projects could still affect the analysis results. Note also that all IFPUG measurements prior to release 4.0 were excluded, tough some suggestions exist to translate the project sized with IFPUG 3.4 into 4.0 measures by a conventional average coefficient.

Step	Filtering Variable	Filtering Criteria	Excluded Records	Residual Records	Residual Percent
1	PRJ_TYPE	= “New Development” OR “Enhancement”	1 “New Utility”, 1 “Purchased Package”, 55 “Re-Development”	1,970	97.2%
2	MM	= “IFPUG”	195 not “IFPUG”	1,775	87.6%
3	DQR	= “A” OR “B”	80 “C”, 33 “D”	1,662	82.0%
4	FP_STD_PRIMARY	= “IFPUG *”	336 not “IFPUG *”	1,326	65.4%
Ä	**Sample A (“soft filter” sub-sample; 1,326 records)**				
5	PACKAGE	≠ “Y”	68 “Y”	1,258	62.1%
6	UFP_RAT	= “A” OR “B”	184 “C”, 1 “D”	1073	52.9%
7	FP_STD_PRIMARY	= “IFPUG 4.*”	1 “IFPUG” (1993), 13 “IFPUG 2”, 136 “IFPUG 3”, 9 “IFPUG 3.4”	914	45.1%
8	RL	= “1” OR “2”	6 “3”, 134 “4”	774	38.2%
Ä	**Sample B (“severe filter” sub-sample; 774 records)**				

Table 2: Sequence of applied filters. Starting N is 2,027; the residual percent on every row is referred to this original amount.

Next, Table 3 reports a brief list of definitions for some values, as provided by ISBSG, that were taken into account when deciding the filtering thresholds.

Variable	Definition	Value	Value Definition
DQR	Data Quality Rating This field contains an ISBSG rating code, applied to the project data by the ISBSG quality reviewers.	A	The data submitted was assessed as being sound with nothing being identified that might affect its integrity.
		B	The submission appears fundamentally sound, but there are some factors which could affect the integrity of the submitted data.
		C	Due to significant data not being provided, it was not possible to assess the integrity of the submitted data.
		D	Due to one factor or a combination of factors, little credibility should be given to the submitted data.
UFP_RAT	UFP Rating This field contains an ISBSG rating code applied to the unadjusted FP data by the ISBSG quality reviewers.	A	The unadjusted FP was assessed as being sound with nothing being identified which might affect its integrity.
		B	Whilst assessed as being sound, integrity cannot be assured as a single figure was provided or has been calculated from adjusted FP and VAF provided.
		C	Due to unadjusted FP or count breakdown data not being provided, it was not possible to provide the unadjusted FP data.
		D	Due to one factor or a combination of factors, little credibility should be given to the unadjusted FP data provided.
RL	Resource Level About the people whose time is included in the work effort data reported. [A] number means that all support at that and preceding levels is included.	1	Development team effort (e.g., project team, project management, project administration).
		2	Development team support (e.g., database administration, data administration, quality assurance, data security, standards support, audit & control, technical support).
		3	Computer operations involvement (e.g., software support, hardware support, information center support, computer operators, network administration).
		4	End users or clients (e.g., user liaisons, user training time, application users and/or clients).

Table 3: Value definitions for some ISBSG-assigned variables.

3.2 Data categorization

Due to the wide variety of value instances, two variables have been “translated” into new category variables (or “classes”), in order to permit more general distribution analyses and more significant correlation analyses against the other selected variables from the sub-samples. The variables being categorised are:

- IMPL_DATE (Implementation Date), re-aggregated into IMPL_PERIOD (Implementation Period); actually, it was chosen a 2-year period scale, based on the actual distribution of projects over the years) – selected periods are non-overlapping: 1989-1990, 1991-1992, 1993-1994, …, 1999-2000, 2001-2002. IMPL_PERIOD is an indicator of “when the project was completed”, not of “when the project was executed” – e.g., the implementation period of a project, executed during 2000, but implemented in January 2001, is "2001-2002", not 1999-2000. Further reasoning, based on the original "Project Elapsed Time" (in months, from the original ISBSG sample, provided in approx. 80% of the records), could lead to an improved classification variable.

- PRIM_PROG_LANG (Primary Programming Language), re-aggregated into LANG_LEV (Language Level), and further re-aggregated into LL_CAT (Language Level Category); Capers Jones' well-known programming languages table was taken as a reference for such classification, since the basic 3GL/4GL/5GL classification of programming languages looked rather uncertain in terms of assignation and too generic to highlight significant differences among languages, e.g. for productivity analyses. Table 4 (below) reports the language level categories and the specific level of most frequent languages in the samples, according to Capers Jones' approach.

LL_CAT	LANG_LEV Range	Examples
LL_CAT 1	1-3	ASSEMBLER (1); C (2.5); COBOL, COBOL 2, MVS COBOL (3); FORTRAN (3); PASCAL (3.5).
LL_CAT 2	4-8	PL/I (4); Generic 3rd Generation Language (4); LISP (5); C++, JAVA (6); ADA (6.5); CICS (7); ORACLE (8); MS ACCESS (8.5).
LL_CAT 3	9-15	VISUAL BASIC, VBSCRIPT (9); VISUAL C++ (9.5); DELPHI (11); PRO C (12); COOL.GEN (14); LOTUS NOTES, SMALLTALK, UNIX SHELL SCRIPT (15).
LL_CAT 4	16-23	ADS/O, NOTES SCRIPT, Generic 4th Generation Language (16); CLIPPER (17); POWERBUILDER, TELON, ABAP, SAP ABAP (20); HTML (22); ASP (23).
LL_CAT 5	24-55	SQL, EASYTRIEVE (25); PL/SQL, SQL WINDOWS (27); Spreadsheet (50).
LL_CAT 6	>55	Generic 5th Generation Language (70)

Table 4: Programming languages (examples), levels and level categories.

3.3 Data transformation

The only relevant data transformation taken by the SBC was:

- UFP size attribute equalized to the (adjusted) FP values from the ISBSG database for those projects, where only "FP", with no "VAF" and no "function breakdown detail", was provided; analysing further data only by means of *size ranges* – see next section 4 – can smooth the risk carried by this hypothesis;
- PDR re-calculated by SBC, including previously void values were UFP is achieved (previous item).

No statistical normalization of numerical variables was performed.

3.4 Values distributions in final sub-samples

Sample A ("soft filter" sub-sample) and sample B ("severe filter" sub-sample) were obtained through the previously depicted filtering, categorization and transformation actions; they contain, respectively, 65.4% and 38.2% of the original ISBSG database recordset. The sub-samples characteristics are reported in the following Tables 5 and 6, where:

- "DEV" and "ENH" stand, respectively, for "(new) development" and "enhancement" (project type) - "Aggr." stands for "Aggregated";
- "PRJ_ID" is omitted, but still present for sake of traceability of records;
- percentages are referred to each sub-samples, by column, not to the overall database;
- aspects that remained critical are highlighted in grey (aggregation is not considered a critical aspect, since it tends to smooth single suspect values into wider categories);
- aspects that were improved are highlighted with bold, italic text style.

Note that no variable in these sub-samples has multiple values, as in the original overall ISBSG database.

Variable	N	%	N_{DEV}	$\%_{DEV}$	N_{ENH}	$\%_{ENH}$	Value Type	Value Range	Calc
UFP	1,326	[100%]	588	[100%]	738	[100%]	Num.	-	Part.
PLATFORM	887	66.9%	430	73.1%	457	61.9%	Text	3	
LANG_LEV	1,052	79.3%	468	79.6%	584	79.1%	***Ord.***	***29***	Aggr.
LANG_CAT	1,052	79.3%	468	79.6%	584	79.1%	***Ord.***	***6***	Aggr.
IMPL_PERIOD	1,215	***91.6%***	540	91.8%	675	91.5%	***Ord.***	***7***	Aggr.
WE_TOT	1,326	***100.0%***	588	100.0%	738	100.0%	Num.	-	
PDR	1,326	***100.0%***	588	100.0%	738	100.0%	Num.	-	Yes

Table 5: Characteristics of selected variables in sample A ("soft filter").

Variable	N	%	N_{DEV}	$\%_{DEV}$	N_{ENH}	$\%_{ENH}$	Value Type	Value Range	Calc
UFP	774	[100%]	299	[100%]	475	[100%]	Num.	-	Part.
PLATFORM	386	49.9	168	56.2%	217	45.7%	Text	3	
LANG_LEV	587	75.8	214	71.6%	373	78.5%	***Ord.***	***29***	Aggr.
LANG_CAT	587	75.8	214	71.6%	373	78.5%	***Ord.***	***6***	Aggr.
IMPL_PERIOD	719	***92.9***	270	90.3%	449	94.5%	***Ord.***	***6***	Aggr.
WE_TOT	774	***100.0%***	299	100.0%	475	100.0%	Num.	-	
PDR	774	***100.0%***	299	100.0%	475	100.0%	Num.	-	Yes

Table 6: Characteristics of selected variables in sample B ("severe filter").

4 Distribution analyses

Simple distribution analyses are reported for selected variables, in analogy with those of section 2.3, over the specific sub-samples A and B. As obviously expected, that project size, effort and (consequently) delivery rate distributions remain skewed towards low values. Since the sub-samples contain only specific measurement methods, significant hypotheses can be made about such distributions; particularly, we can test their log-normality, even by a visual approach. Distribution values are calculated for both the samples, to show how the "quantity versus quality" choice could affect the analyses results.

4.1 Size distributions

UFP (unadjusted function point size – measurement unit: UFP)

	N	%	Min	P10	P25	Median	P75	P90	Max	Mean	Std Dev
Sample A											
DEV	588	44.3%	6.0	109.0	186.0	379.0	857.0	1,741.6	16,148.0	787.5	1,369.0
ENH	738	55.7%	6.0	56.7	96.3	174.0	378.8	782.5	20,000.0	439.5	1,272.7
TOT	1,326	100%	6.0	67.0	119.0	248.5	550.0	1,266.0	20,000.0	593.8	1,327.0
Sample B											
DEV	299	38.6%	25.0	112.0	174.5	***334.0***	676.0	1,438.4	16,148.0	666.5	***1,227.8***
ENH	475	61.4%	6.0	56.0	88.0	***153.0***	281.0	604.4	7,134.0	282.2	***470.4***
TOT	774	100%	6.0	63.0	108.3	***201.5***	429.8	928.3	16,148.0	430.6	***867.1***

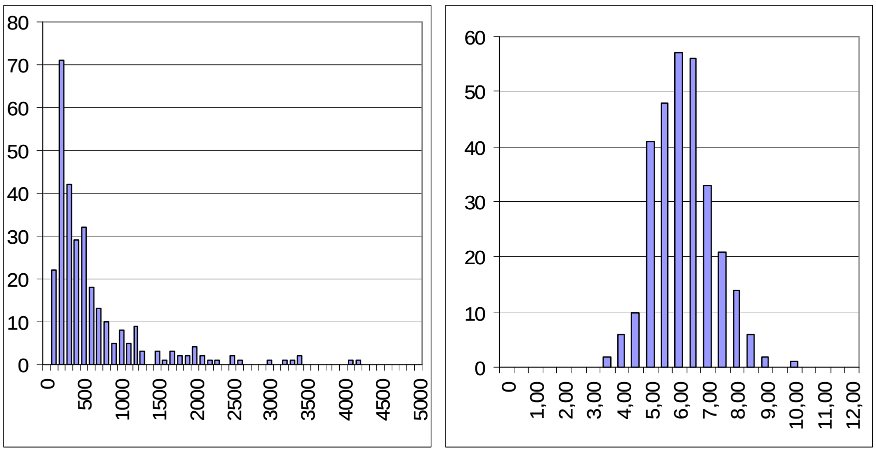
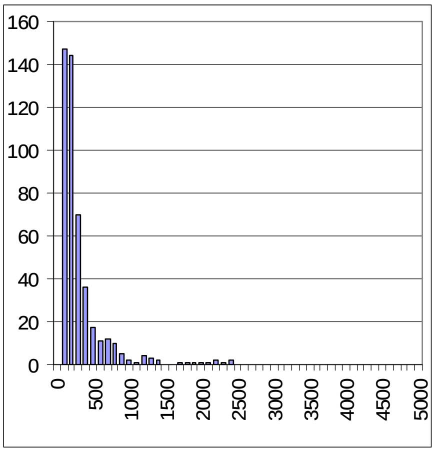


Figure 1: Linear (left) and logarithmic (right) distributions for development projects sizes (examples from Sample B – "severe filter", N = 299).

From the analysis of percentiles on the logarithmic distribution for development projects (Figure 1), a limited set of development project *size classes* is obtained, as reported in Table 7. Such classes are used for categorization of subsequent two-variable analyses; they are proposed for standardized use in data exchange, agreement definitions, software macro-estimation approaches. Similarly, size classes are obtained for enhancement projects (Figure 2 and Table 8).

Code	SIZE_CLASS	Size Range (UFP)
DEV_{XS}	Very Small	0-150
DEV_{S}	Small	150-300
DEV_{M}	Medium	300-600
DEV_{L}	Large	600-1,200
DEV_{XL}	Very Large	*1,200-5,000*
DEV_{XXL}	Extremely Large	> 5,000

Table 7: Development projects size classes.



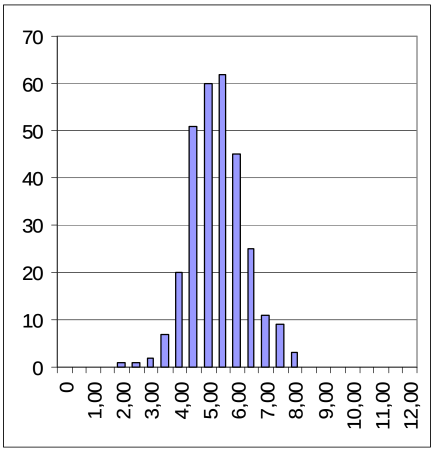


Figure 2: Linear (left) and logarithmic (right) distributions for enhancement projects sizes (examples from Sample B – "severe filter", N = 475).

Code	SIZE_CLASS	Size Range (UFP)
ENH_{XS}	Very Small	0-60
ENH_{S}	Small	60-120
ENH_{M}	Medium	120-240
ENH_{L}	Large	240-480
ENH_{XL}	Very Large	*480-2,000*
ENH_{XXL}	Extremely Large	> 2,000

Table 8: Enhancement projects size classes.

4.2 Effort distributions

WE_TOT (summary work effort – measurement unit: ph)

	N	%	Min	P10	P25	Median	P75	P90	Max	Mean	Std Dev
Sample A											
DEV	588	44.3%	17.0	620.9	1,177.0	3,060.0	7,501.3	17,796.9	164,764.0	7,612.8	13,929.5
ENH	738	55.7%	90.0	405.1	779.8	1,788.0	4,425.8	9,901.3	100,529.0	4,592.1	8,547.3
TOT	1,326	100%	17.0	461.0	973.0	2,311.0	5,753.0	14,524.0	164,764.0	5,931.6	11,351.2
Sample B											
DEV	299	38.6%	50.0	597.6	1,057.5	***2,540.0***	5,924.0	14,911.8	73,920.0	6,232.1	***10,593.9***
ENH	475	61.4%	90.0	426.0	762.5	***1,642.0***	3,911.0	8,245.4	53,830.0	3,560.2	***5,607.1***
TOT	774	100%	50.0	455.5	867.5	***1,913.5***	4,713.5	10,023.3	73,920.0	4,592.3	***8,014.9***

No specific consideration are reported for effort distributions. More significant information is expected from the project delivery rate, i.e. summary work effort normalised by the size (next distribution set).

4.3 Productivity distributions

PDR (project delivery rate – measurement unit: ph/UFP)

	N	%	Min	P10	P25	Median	P75	P90	Max	Mean	Std Dev
Sample A											
DEV	588	44.3%	0.1	2.0	3.9	8.3	16.8	28.5	640.0	14.3	31.7
ENH	738	55.7%	0.1	2.3	4.8	10.7	21.7	41.0	327.4	18.3	25.5
TOT	1326	100%	0.1	2.1	4.3	9.7	19.0	35.0	640.0	16.5	28.5
Sample B											
DEV	299	38.6%	0.1	2.1	3.9	***8.0***	16.2	23.1	300.3	12.8	***21.5***
ENH	475	61.4%	0.3	2.5	4.8	11.0	23.0	44.6	327.4	19.8	29.1
TOT	774	100%	0.1	2.3	4.3	***9.5***	19.3	36.0	327.4	17.1	***26.6***

As for the size and effort distribution, a skewed distribution is observed for project delivery rate. A log-normal distribution could be considered (Figures 3, 4, for Sample B), but a cut off is present, that requires further analysis.

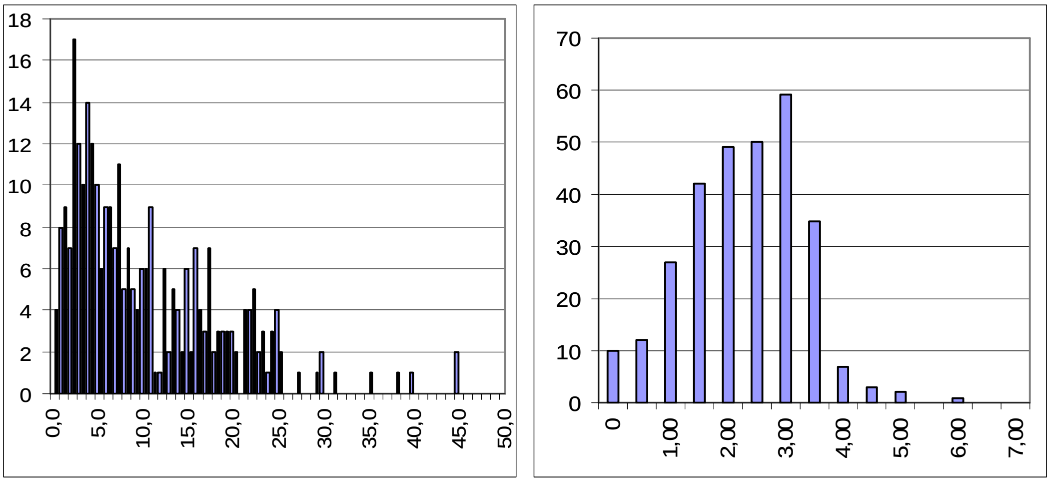


Figure 3: Linear (left) and logarithmic (right) distributions for development projects PDRs' (examples from Sample B – "severe filter", N = 299).

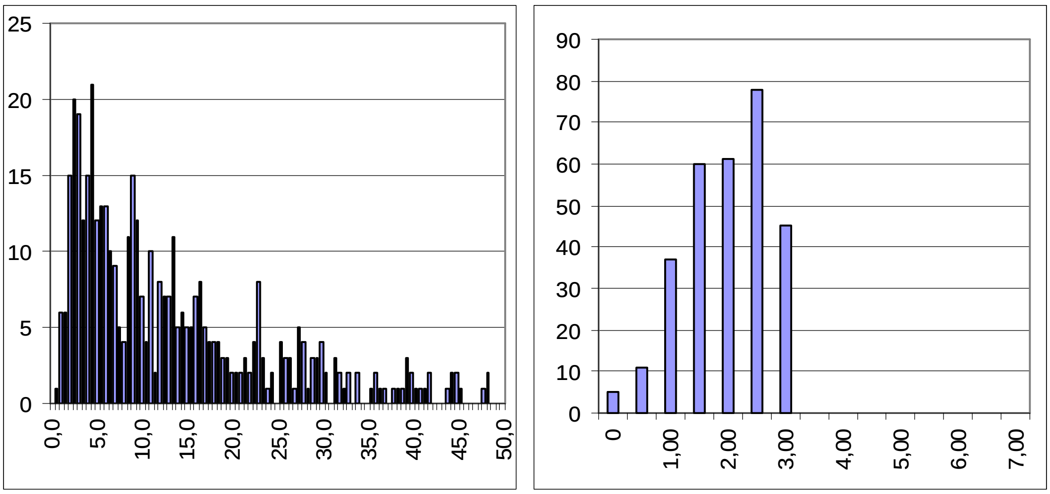


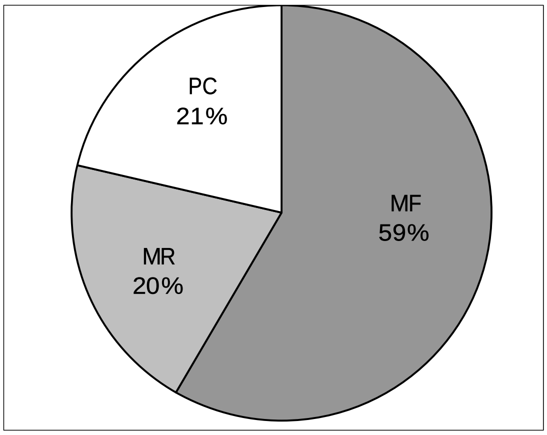
Figure 4: Linear (left) and logarithmic (right) distributions for enhancement projects PDRs' (examples from Sample B – "severe filter", N = 475).

4.4 Platform distributions

PLATFORM (development platform – textual variable)

	N	%	MF	%_N	MR	%_N	PC	%_N	Check
	Sample A								
DEV	430	48.5%	242	56.3%	74	17.2%	114	26.5%	100%
ENH	457	51.5%	390	85.4%	34	7.4%	33	7.2%	100%
TOT	887	100%	632	71.2%	108	12.2%	147	16.6%	100%
	Sample B								
DEV	168	43.6%	98	58.3%	34	20.3%	36	21.4%	100%
ENH	217	56.4%	196	90.3%	9	4.2%	12	5.5%	100%
TOT	385	100%	294	76.3%	43	11.2%	48	12.5%	100%

Legenda: MF = Mainframe, MR = Midrange, PC = Personal Computer



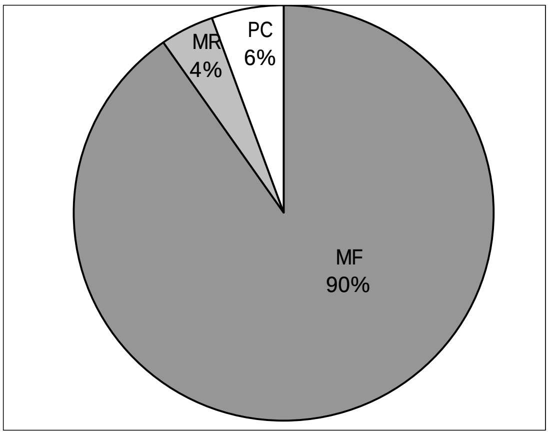


Figure 5: Development (left) and enhancement (right) distributions for platform (examples from Sample B – "severe filter").

Although the platform distributions are clear enough, we found some difficulty in interpreting the assignment of such values (e.g. "midrange" versus "personal computer" for some application types). We suggest that a wider or cleaner range of values could provide a more effective analysis variable. The platform distribution of enhancement projects is highly dominated by Mainframe cases (Figure 5).

4.5 Language level category distributions

LL_CAT (language level category – ordinal variable)

	N	%	LLC1	%_N	LLC2	%_N	LLC3	%_N	LLC4	%_N	LLC5	%_N	LLC6	%_N	Check
Sample A															
DEV	468	44.5%	175	37.4%	112	23.9%	63	13.5%	80	17.1%	38	8.1%	0	0%	100%
ENH	584	55.5%	274	46.9%	197	33.7%	45	7.7%	50	8.6%	17	2.9%	1	0.2%	100%
TOT	1052	100%	449	42.7%	309	29.4%	108	10.3%	130	12.4%	55	5.2%	1	0.1%	100%
Sample B															
DEV	214	36.5%	98	45.8%	38	17.8%	28	13.1%	28	13.1%	22	10.3	0	0%	100%
ENH	373	63.5%	195	52.3%	102	27.3%	34	9.1%	27	7.2%	14	3.8%	1	0.3%	100%
TOT	587	100%	293	49.9%	140	23.9%	62	10.6%	55	9.4%	36	6.1%	1	0.2%	100%

For both development and enhancement project types, and both samples, the Language Level Category distribution has the maximum at LLC1 (Language Level 1-3), followed by LLC 2 (Language Level 4-8). A more specific analysis (not reported here) shows a peak for Level 3 languages (mostly COBOL).

Both samples present many projects for which the Primary Programming Language is not provided. Since the Programming Language represents an important influence factor for Project Delivery Rate, we strongly suggest that such information should be necessarily provided for mostly records.

4.6 Implementation period distributions

IMPL_PERIOD (implementation period – ordinal variable)

	N	%	1989-90	1991-92	1993-94	1995-96	1997-98	1999-2000	2001-02
Sample A									
DEV	540	44.4%	13	44	104	97	143	125	14
ENH	675	55.6%	2	19	60	88	184	159	163
TOT	1,215	100%	15	63	164	185	327	284	177
Sample B									
DEV	270	37.6%	0	2	42	41	106	73	6
ENH	449	62.4%	0	0	10	24	163	95	157
TOT	719	100%	0	2	52	65	269	168	163

No specific distribution is expected for implementation period. Further data collection is expected to augment the recent periods percentages.

5 Two-variables analyses

Two-variables analysis is introduced to investigate the dependency of PDR against other variables in a visual approach. Any numerical regression analysis is intentionally omitted, at this moment, to avoid a potential misuse ot the analysis results. The subsequent analysis of PDR are a specialization of its distribution over the ranges of other variables (e.g. project size, language level category, implementation period).

5.1 PDR vs SIZE_CLASS

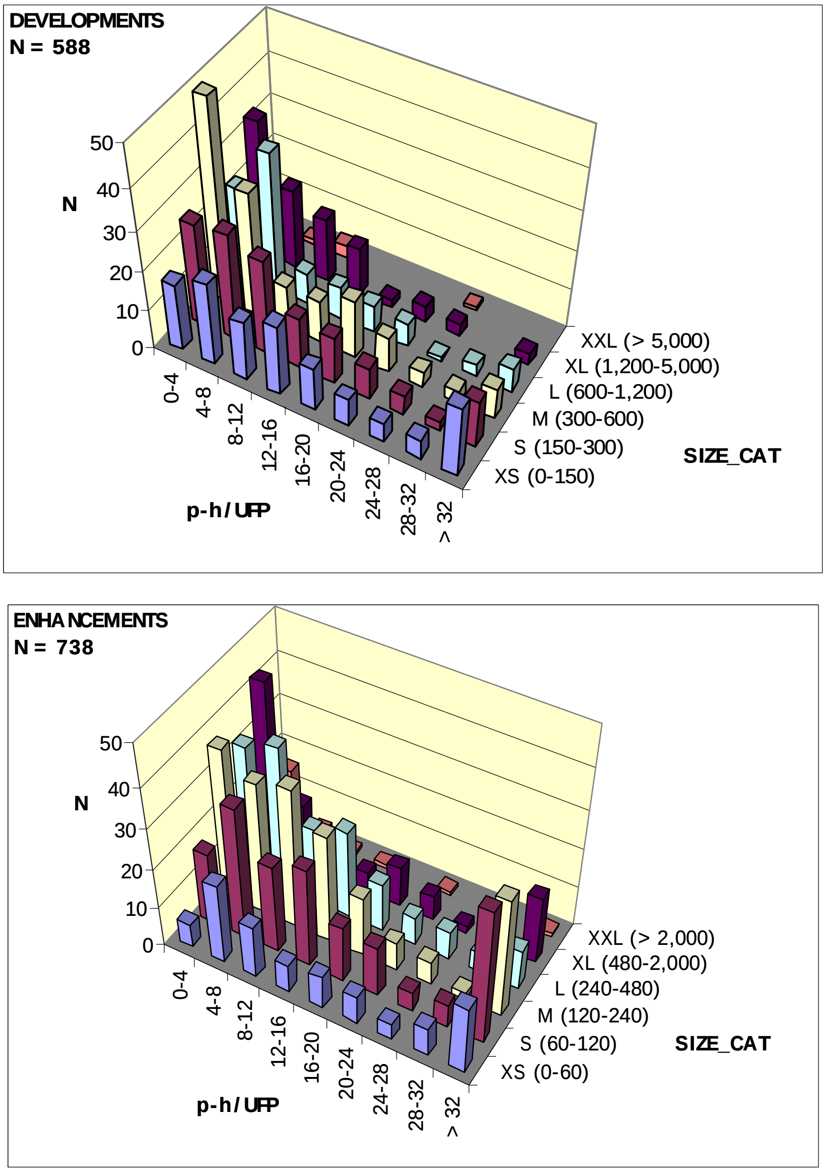


Figure 6: PDR distributions against Size Class (Sample A – “soft filter”).

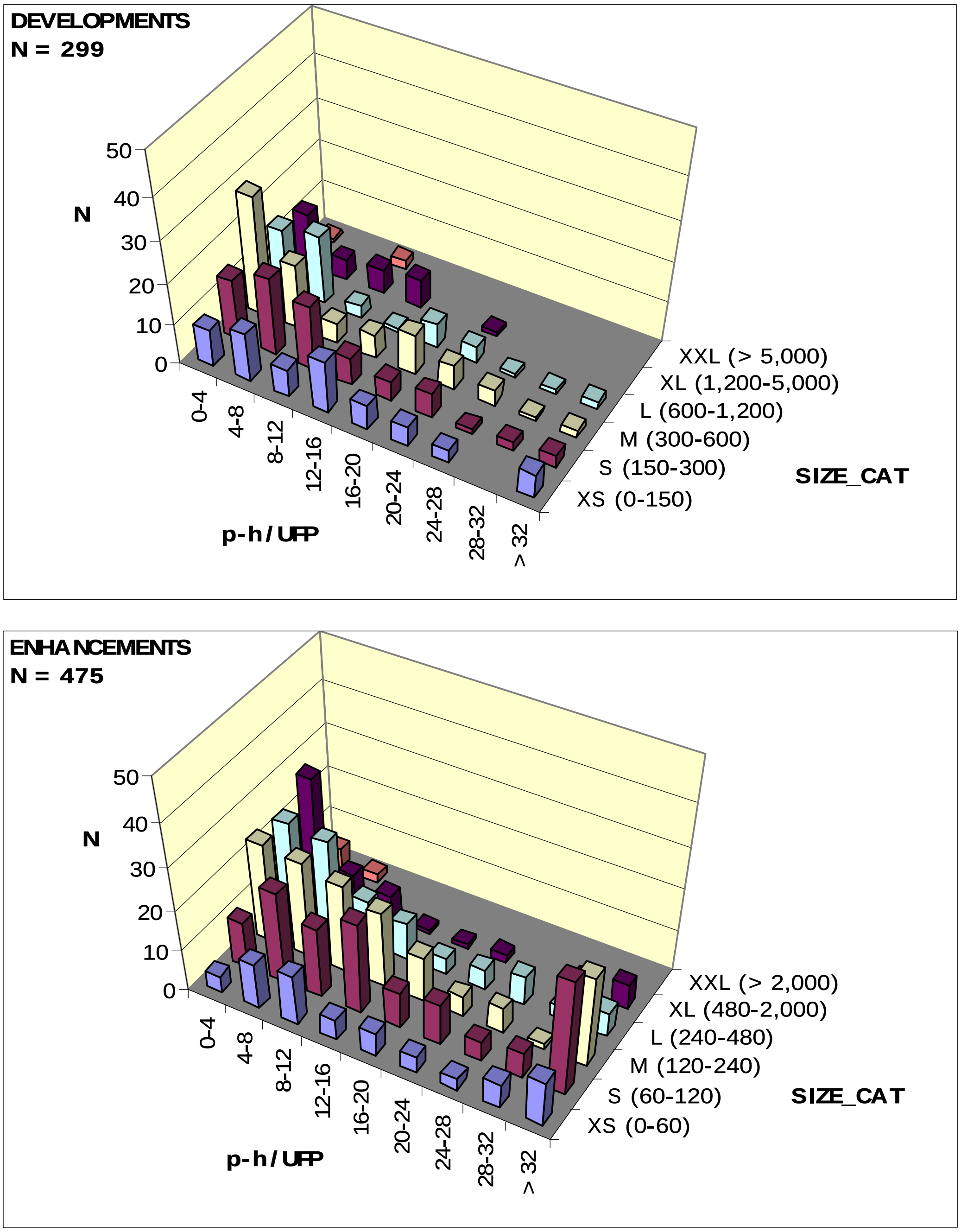
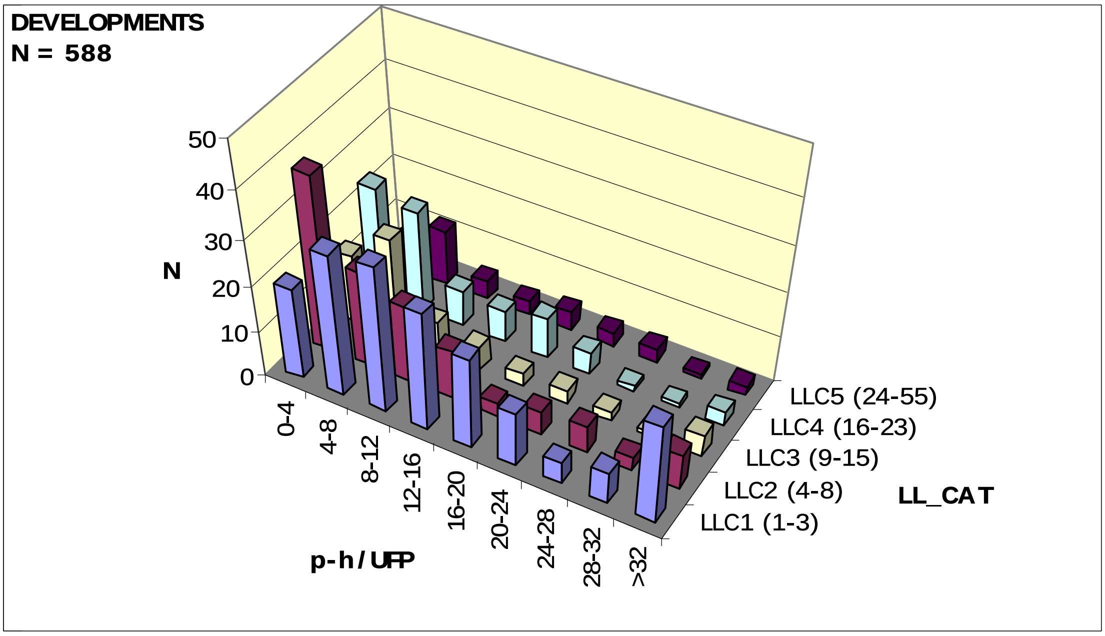


Figure 7: PDR distributions against Size Class (Sample B – "severe filter"). Although some log-normal trends could be perceived, more data in the "severe filter" sample could provide more regular distributions of PDR by Size Class. Peak frequency values of PDR are present in the ranges between few person-hours per UFP and the 8-12 PDR range (8 p-h = approx. 1 person-day). Seemingly, the enhancement PDR on the average is not less than development PDR values.

5.2 PDR vs LL_CAT



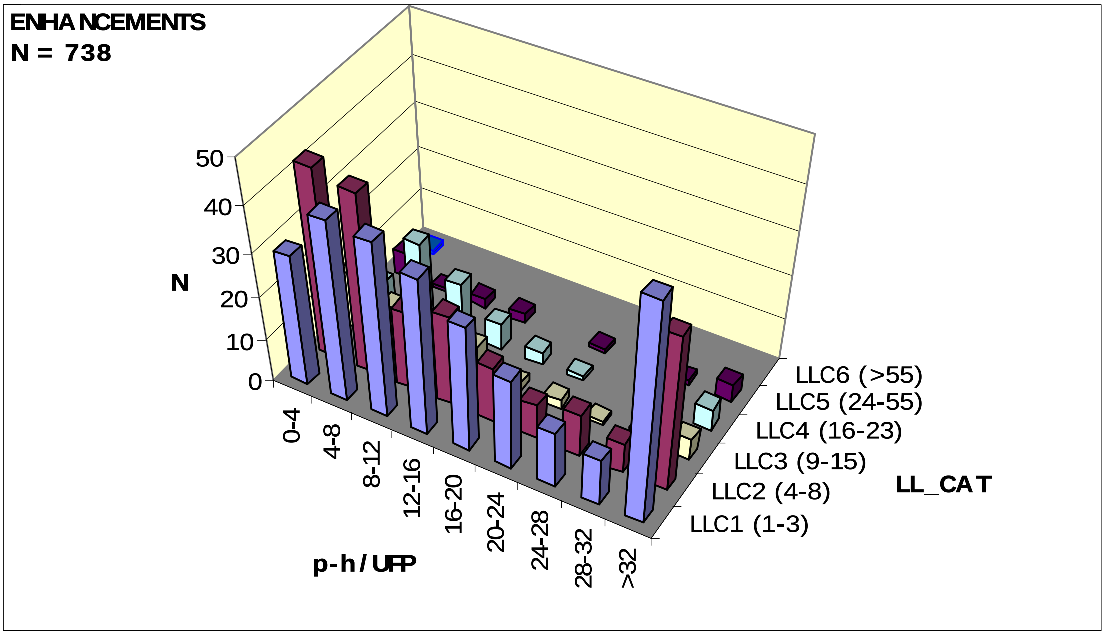


Figure 8: PDR distributions against Language Level Category (Sample A – "soft filter"). For development projects, the PDR range 0-4 (corresponding to the best productivity) includes the most number of projects in the Language Level Class 2 (LL 4-8), including languages such as C++, JAVA, ORACLE. In the same PDR range, the Language Level Class 4 (LL 16-23) is however relevant (including Power Builder, TELON, ADS, HPS, CLIPPER, IDEAL). For enhancement projects, in the PDR range 0-4, the Language Level Class 1 (LL 1-3), including mostly COBOL and C, is also important.

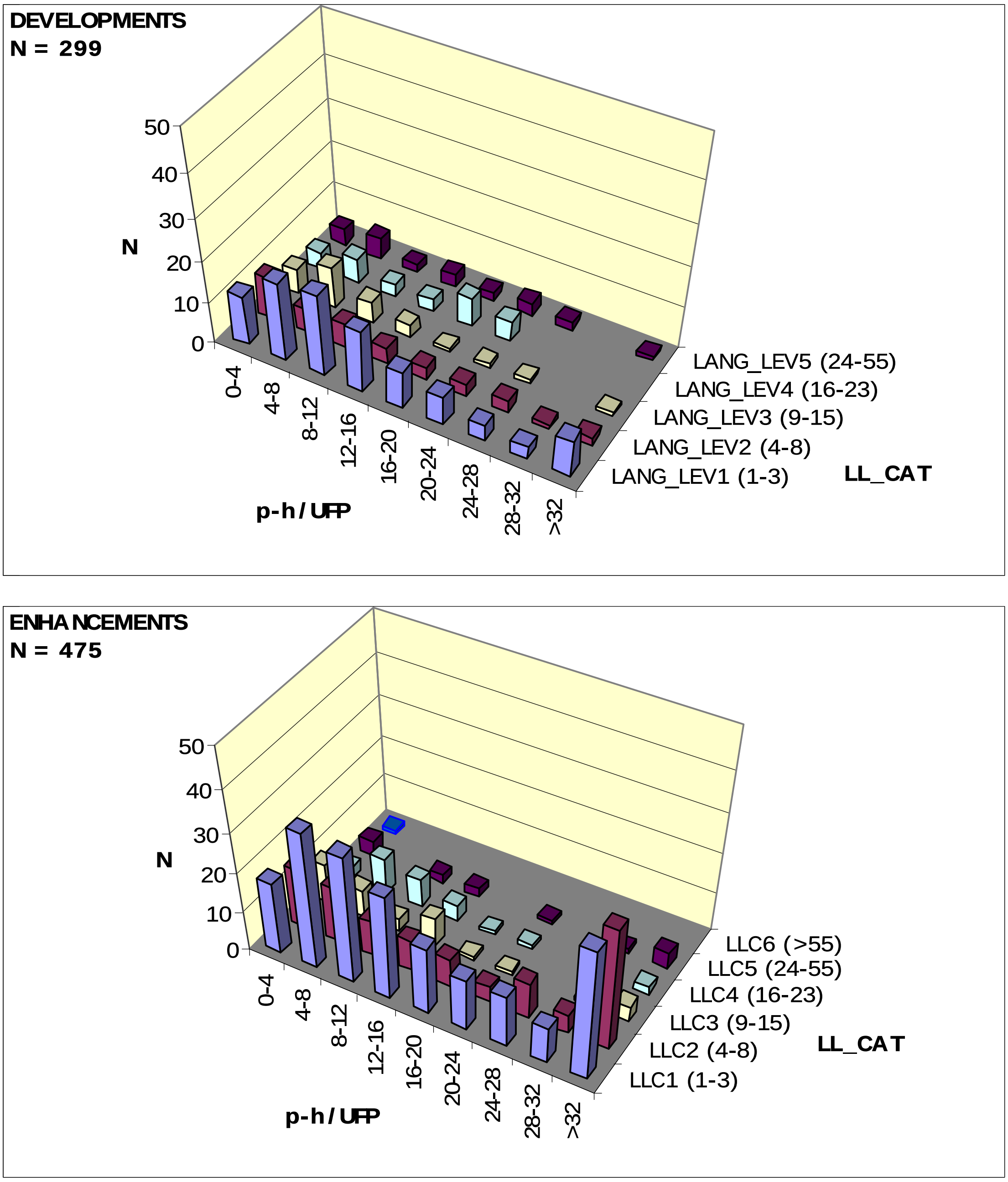
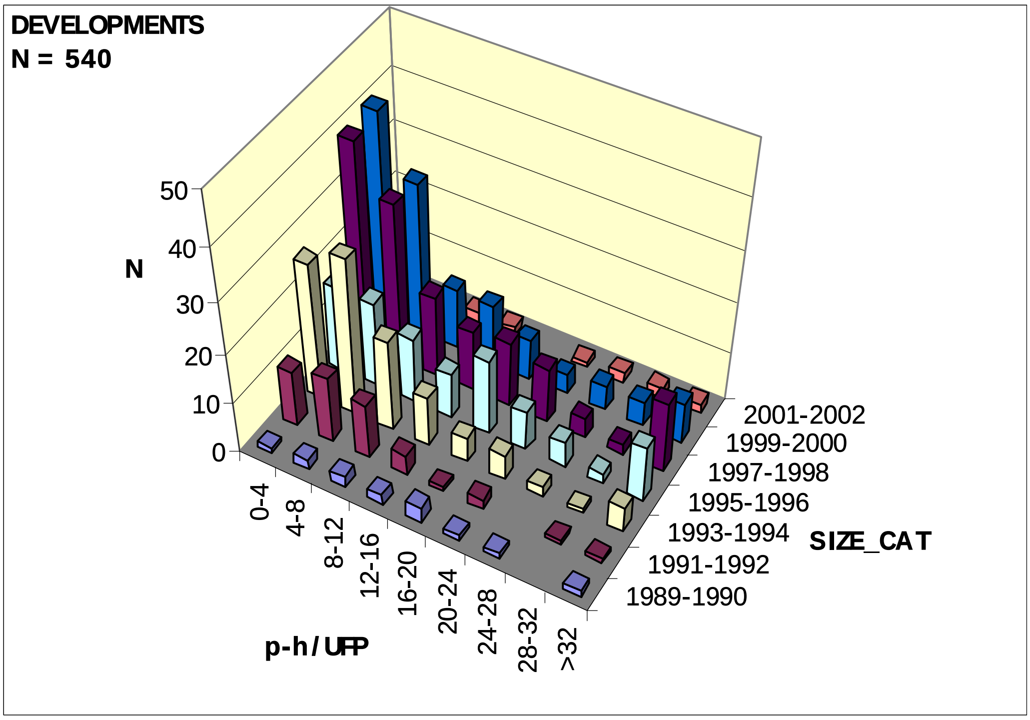


Figure 9: PDR distributions against Language Level Category (Sample B – "severe filter").

For a more effective benchmarking, it is suggested to collect not only the Primary Language, but at least the *Secondary* one (with specified *percentage ratios* of Primary to Secondary languages).

5.3 PDR vs IMPL_PERIOD



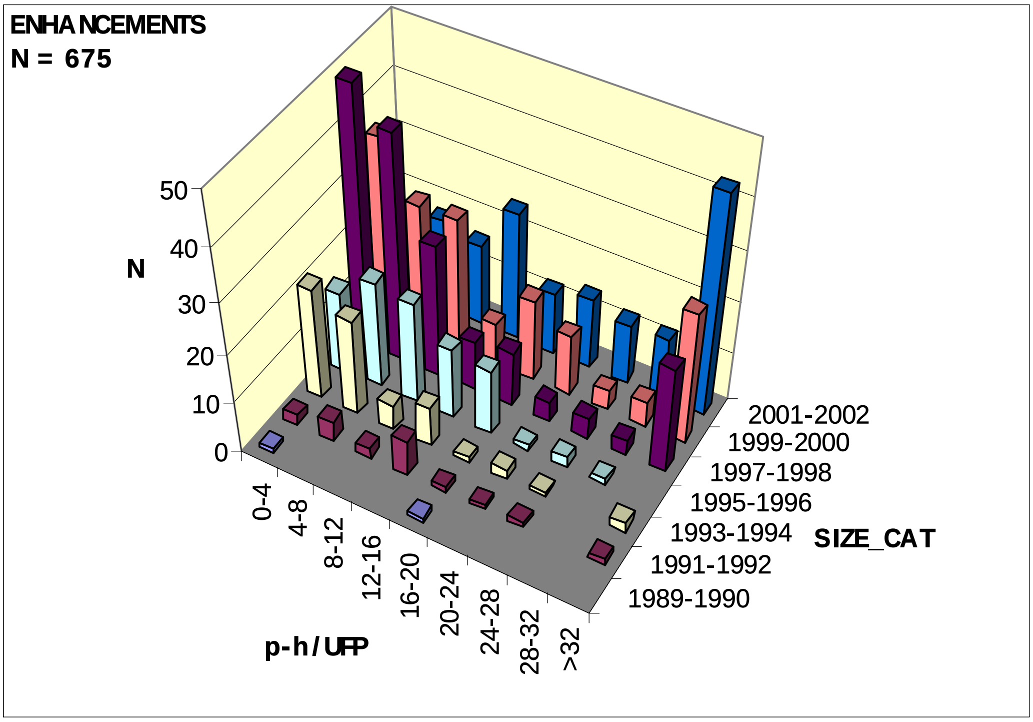
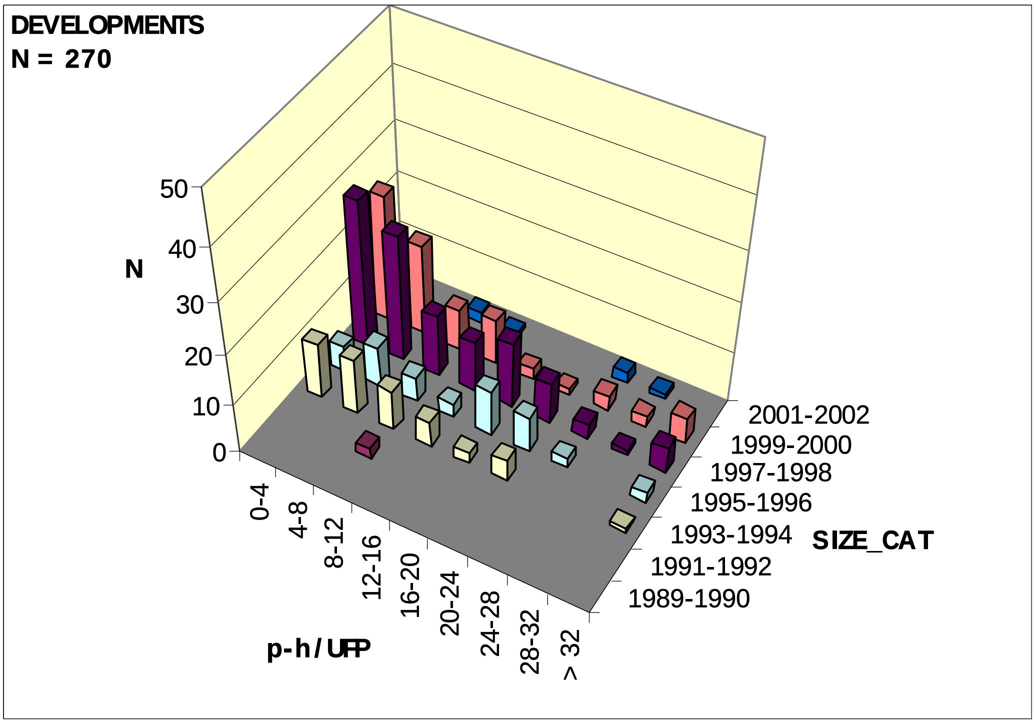


Figure 10: PDR distributions against Implementation Period (Sample a – "soft filter"). The PDR distribution in case of enhancement projects in recent years (2001-2002) shows a peak for PDRs' > 32 p-h/UFP, due to a tail containing several extremely high values (eventually outliers, requiring more investigation).



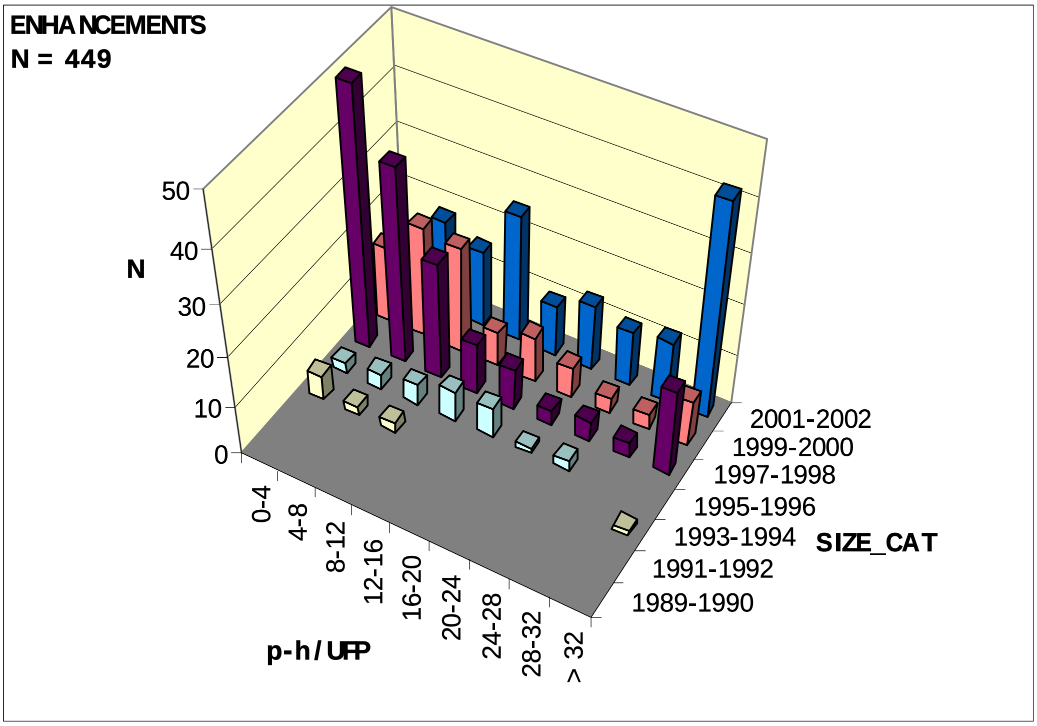


Figure 11: PDR distributions against Implementation Period (Sample B – "severe filter"). Even the "severe filter" sample shows an abnormal peak for recent years (2001-2002), as in case of Sample A (Figure 10).

6 Conclusions

Several hints have been highlighted throughout this first step of benchmarking analysis by the GUFPI-ISMA SBC, and are summarized in the following list:

- dichotomization (to avoid multiple values per record)
- nomenclature (to avoid distinct values for identical instances)
- taxonomy (to avoid open ranges – including a single "other" exception)
- mandatory completeness (to avoid excessive sample reduction by filtering and risky interpretation of void values)
- variable categorization (to avoid excessive variety of instances)

Moreover, the following is suggested to permit a better interpretation of uncertain values:

- a facultative "comment" field per each variable, by the data provider;
- a mandatory "reliability" field per each variable, by the data provider, on a fixed (explained) scale

These suggestions – regarding both the database structure and its contents – may be considered for future improvements of the ISBSG collection process, as well as for the implementation of any local benchmarking database within an organization. It is trivial that changing or introducing new parameters, quality rating criteria or collection process approaches would render the oldest project data in the current benchmarking database. The counterpart gain will be obvious in terms of usability, reduced analysis effort, analysis completeness and significance. The SBC effort aims to suggest an homogeneous approach to the benchmarking data collection process and analysis.

Although some projects are present with measurement method alternative to IFPUG, as COSMIC and MkII approaches, such sample were found too sparse to permit significant analyses. The next issue of the benchmarking database by the ISBSG, which is expected with over 3'000 projects, should open new views on different measurement methods and their application in the statistical analysis. Therefore, further research developments are:

- extension of the analysis to more variables (e.g. functional breakdown by function type and by enhancement operation type, work effort phase breakdown, differentiation by methodology, by software domain, etc.);
- further two-variables and eventually N-variables analyses;
- repetition of analyses on lager ISBSG samples, when published;
- comparison with ISBSG's and other sources' analyses.

References

1. GUFPI-ISMA SBC 2004 Webpage. Online **http://www.gufpi.org/sbc**
2. ISBSG: ISBSG Shared Benchmarking Repository Report, Release 5, March 1998.
3. ISBSG: Estimating, Benchmarking & Research Suite (incorporating the data disk), CD, Release 8, February 2003.
4. ISBSG 2004 Website. Online **http://www.isbsg.org**
5. Jones, C.: Programming Languages Table, Release 8.2, March 1996, SPR.
6. Levine, D.M., Krehbiel, T.C., Berenson, M.L.: Business Statistics: A First Course, 2^{nd} ed., Prentice-Hall, 2000.