Digital Disruption and Graduate Education

Entrepreneurship and Creativity

by

Veljko Milutinović

Life Fellow of the IEEE.
Life Member of ACM SIGARCH.
The CEO of the IPSI Belgrade, Serbia.
Senior Advisor to Maxeler Technologies, US/UK.
Member of the Advisory Board of The Vienna Congress.
Research Director of the MECO.net, Podgorica, Montenegro.
A Founding Member of The Serbian National Academy of Engineering.
Foreign Member of The Montenegro National Academy of Sciences and Arts.
Member and Honorary Treasurer of The Academy of Europe, London, United Kingdom.
Co-Laureate of the 2012-2014 Single IET Premium Award for Computers and Digital Techniques.
A Best Paper Award at IEEE/ACM HICSS-1986 for a Systolic Array with 4096 200MHz GaAs Microprocessors.
His 2019 Opening Keynote at the Annual EMiT Conference of IET Sets R+D Directions for DataFlow in the postFPGA Era.
His Single-Author Book "Surviving the Design of a 200MH RISC Microprocessor" was a Best Seller of IEEE CS PRESS in 1990's.

Department of Computer Science, Indiana University, Bloomington, USA
CSCI-D 490 + CSCI-D 649

Creative Management of Sophisticated R&D Projects in Industry and Academia

Veljko Milutinović

Note: Versions available for both, VLSI and SoftwareEngineering!

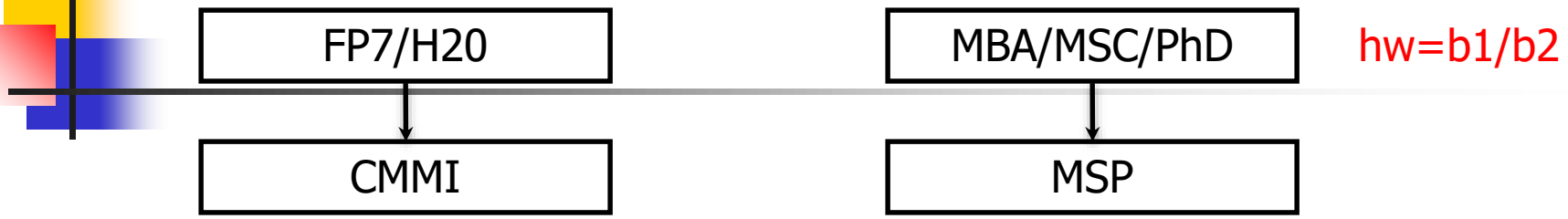
One Good Strategy!

FP7/H20

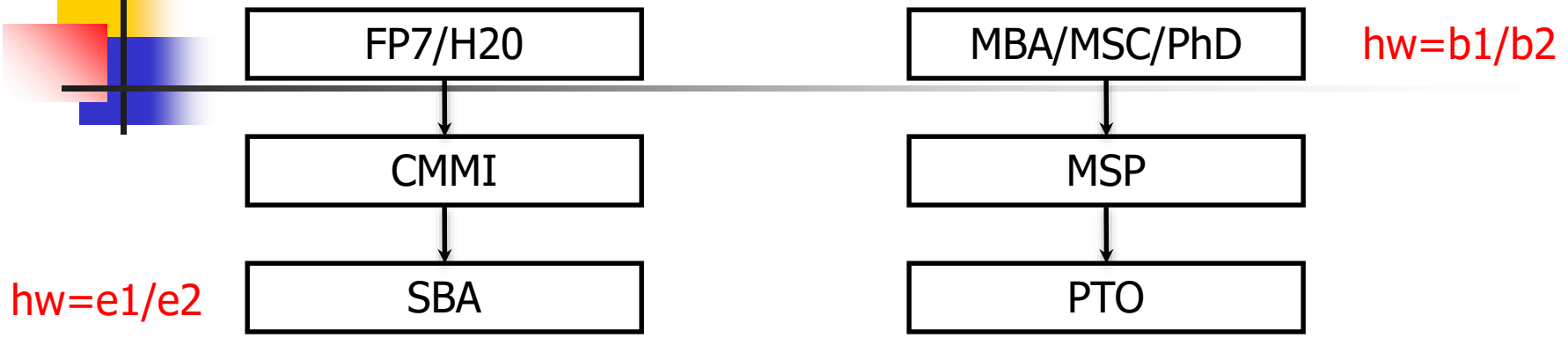
MBA/MSC/PhD

hw=b1/b2

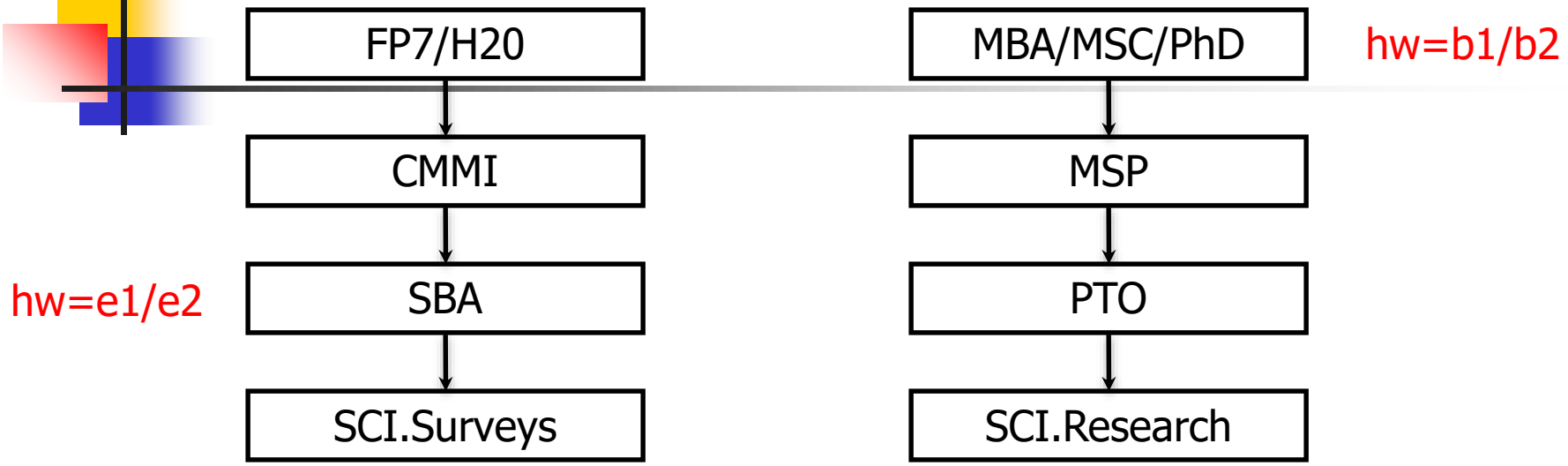
One Good Strategy!



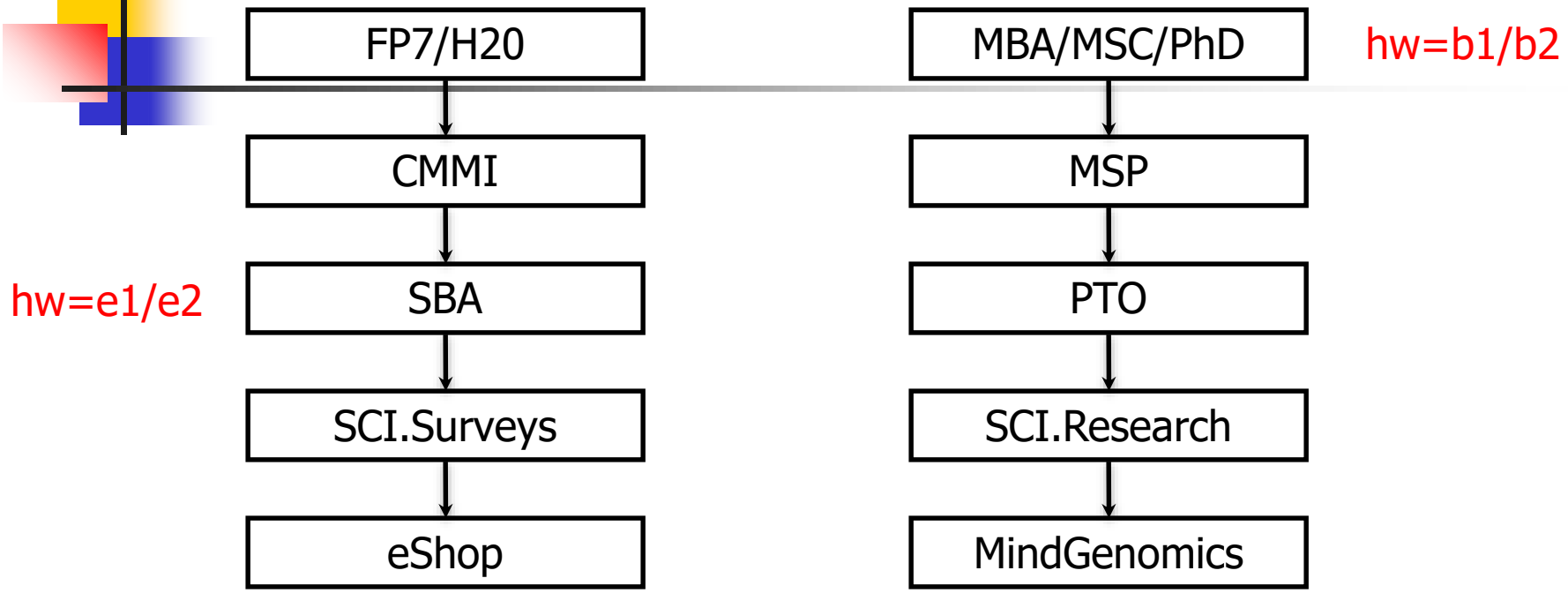
One Good Strategy!



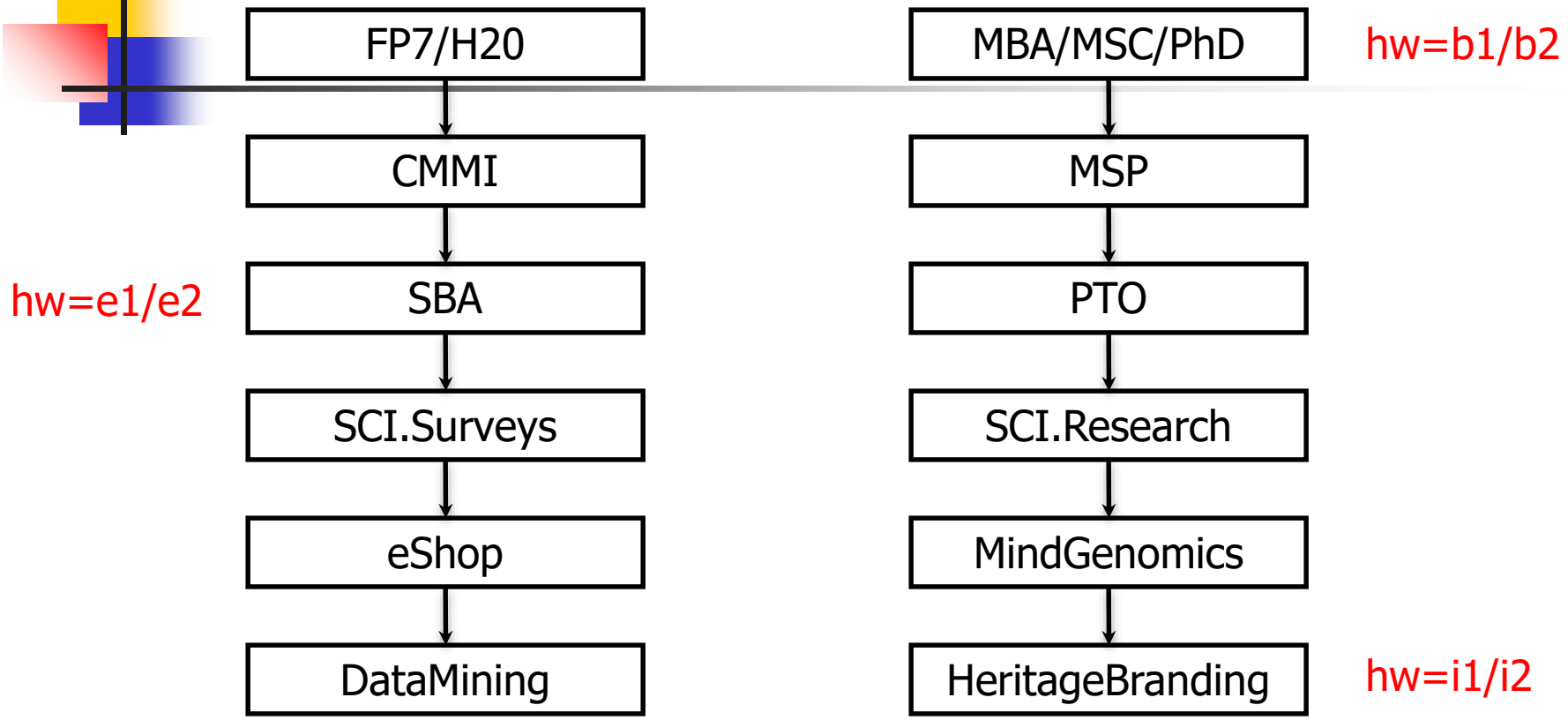
One Good Strategy!



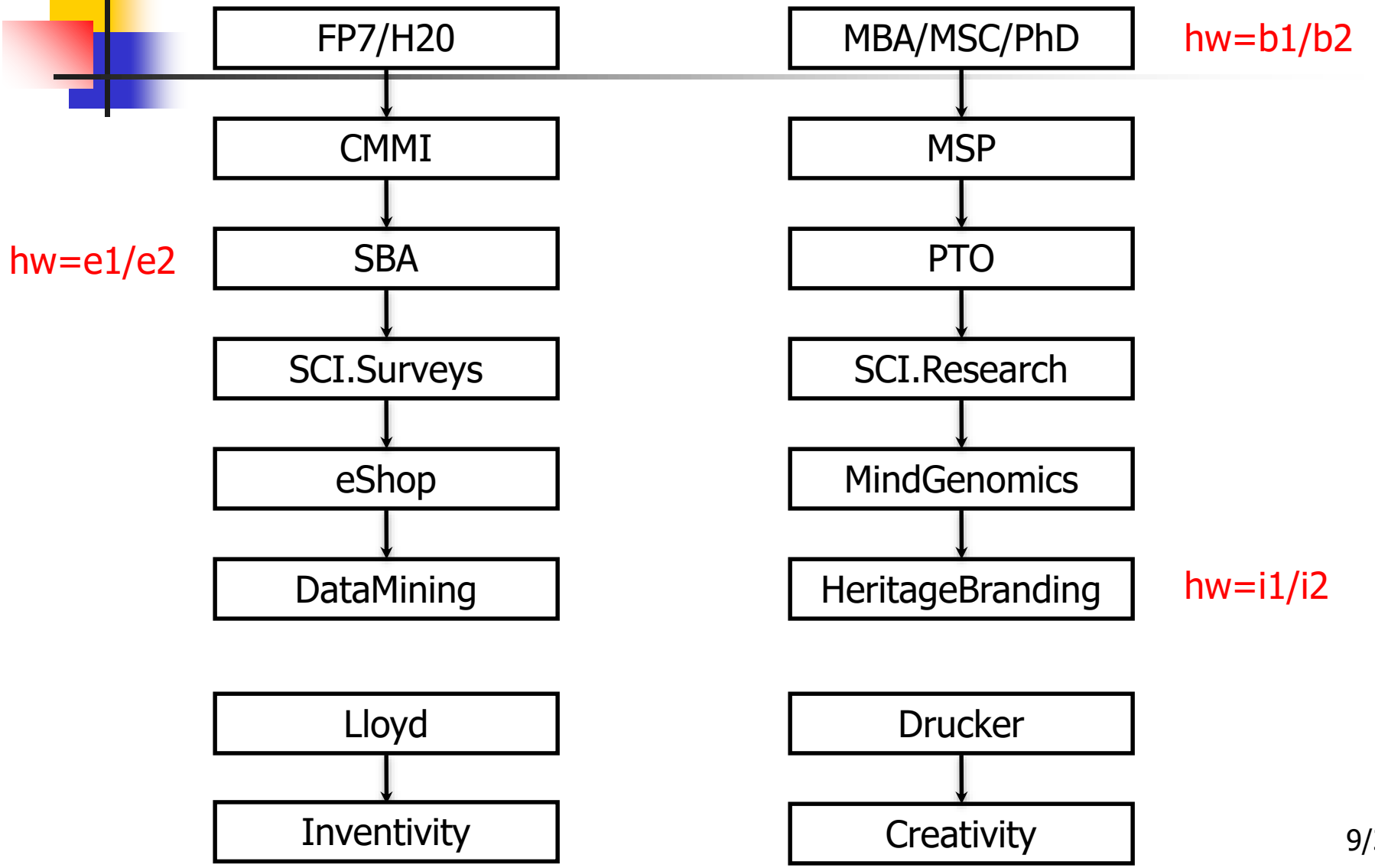
One Good Strategy!



One Good Strategy!



One Good Strategy!



THE "USPI" COURSE OUTLINE:
STAGES OF A COMPLEX RESEARCH/DEVELOPEMENT PROJECT IN ACADEMIA/INDUSTRY

1. **Before the project starts:**
   a. To ensure funding (e.g., H20); HW = H20.doc
   b. To get educated for management (e.g., MBA/PhD); HW = H20.ppt
2. **Soon after the project starts:**
   a. To learn CMMI; HW = CMMI.level2global
   b. To learn Another more focused approach (e.g., Microsoft Project); HW = MP.local
3. **Before the project is over:**
   a. To form a company (e.g., using the SBA guidelines); HW = BizPlan.sba
   b. To protect the company product with a patent (e.g., using the USA patent office guidlines); HW = PatentAplication.pto
4. **Immediately after the project is over:**
   a. To write about existing solutions to the problem (e.g., for a SCI journal); HW = SurveyPaper.doc
   b. To write about obtained results (e.g., for a SCI journal); HW = ResearchPaper.doc
5. **Soon after the project is over:**
   a. To form an Internet shop (small, medium, or large scale); HW = STORE.com
   b. To apply MindGenomics (for better marketing of the result of your project); HW = MicroScienceCustomerTyping.doc
6. **Long after the project is over:**
   a. To apply business intelligence (Data Mining and Media Mining and Lessons Learned); HW = NewAlgorithm.ppt
   b. To brand yourself (Semantic Web and Internet Gallery and Serbia Forum); HW = NewDigitalization.pdf

READINGS:

7. Efficiency
8. Effectiveness
9. Inventivity
10. Creativity
11. Ethics
12. Esthetics

CASE STUDIES:

MENCER/MAXELER
MAURER/MyFORUM
MOSKOWITZ/MindGenomics
MUTLU/CMU/ETH

1: FP/H20/CALLn/ClosingDate/E1M->E4M(STREP)/E40M(IP)

CH (challenges)

15=ST(5)+MF(5)+ED(5)
12
10

↓ 1:1

PP (proposal) ST

↓

OBJ (objectives)

OBJ_1-OBJ_Y [SC (success criteria)$_{1-N}$]

↓ 1:1 ↕

WP (work packages)

WP_0-WP_Y [MS (precise milestones)$_{1-N}$]

↓ 1:M

Tasks

↓ 1:K

Deliverables

If x>=10 and each{ST,MF,ED}>=3
Then: E10K, if leader
E4K, if WP leader
E1K, if partner

RISKS MUST BE ELABORATED! Why?

2: GRE/GMAT (MBA/PHD)

- Verbal reasoning (input), important for teaching schools
- Quantitative reasoning (CPU); tough to compete with Zillions
- Analytical writing (output), important for research schools

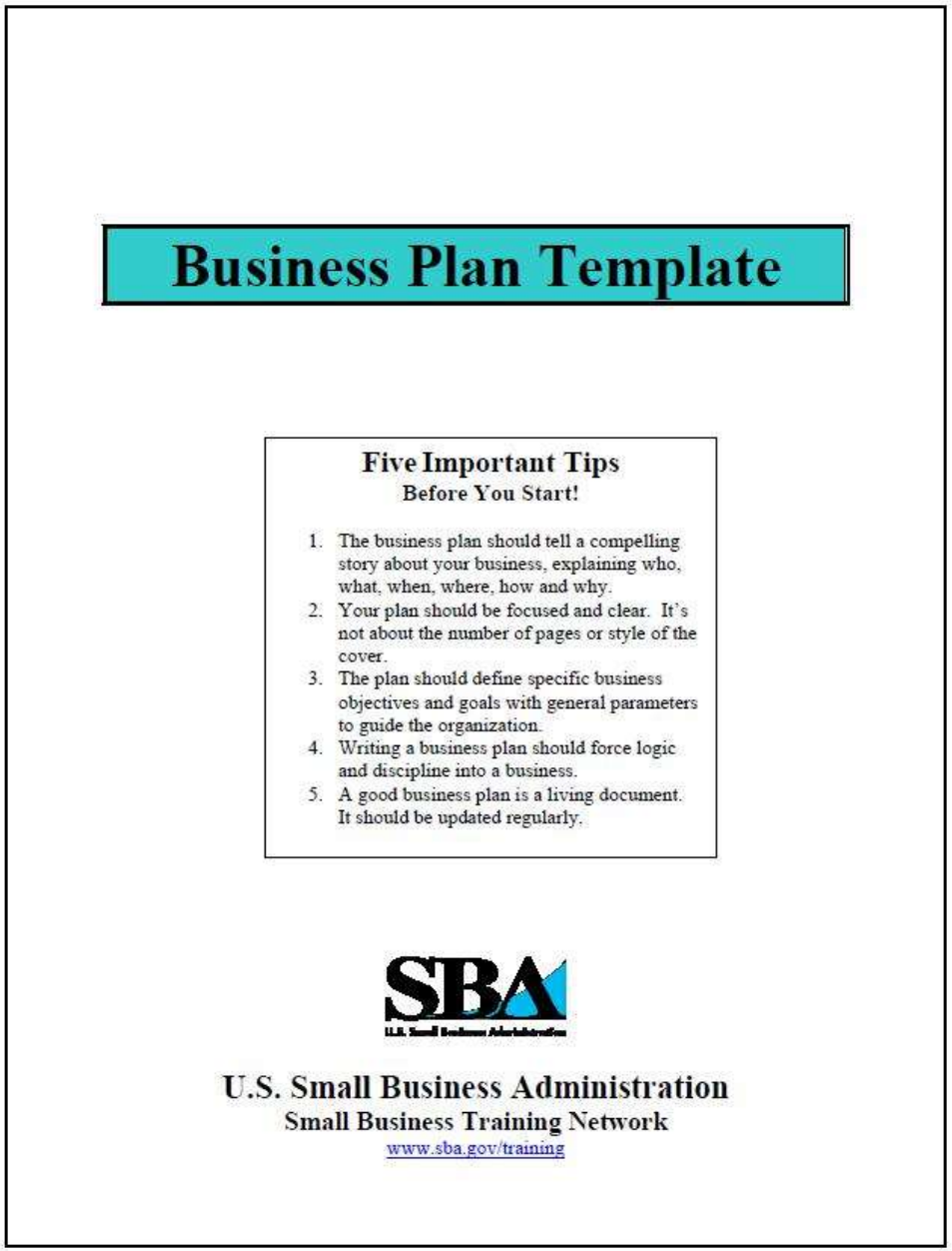
3: Pro Life Management

- The holistic approach: CMMI

4: Pro Life Management

- The specific approach: Agile

5: Harvard Business Plan Template



Business Plan Template

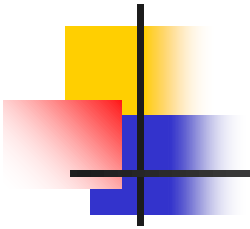
Five Important Tips
Before You Start!

1. The business plan should tell a compelling story about your business, explaining who, what, when, where, how and why.
2. Your plan should be focused and clear. It's not about the number of pages or style of the cover.
3. The plan should define specific business objectives and goals with general parameters to guide the organization.
4. Writing a business plan should force logic and discipline into a business.
5. A good business plan is a living document. It should be updated regularly.

SBA

U.S. Small Business Administration
Small Business Training Network
www.sba.gov/training

6: US Patent and Trademark Office

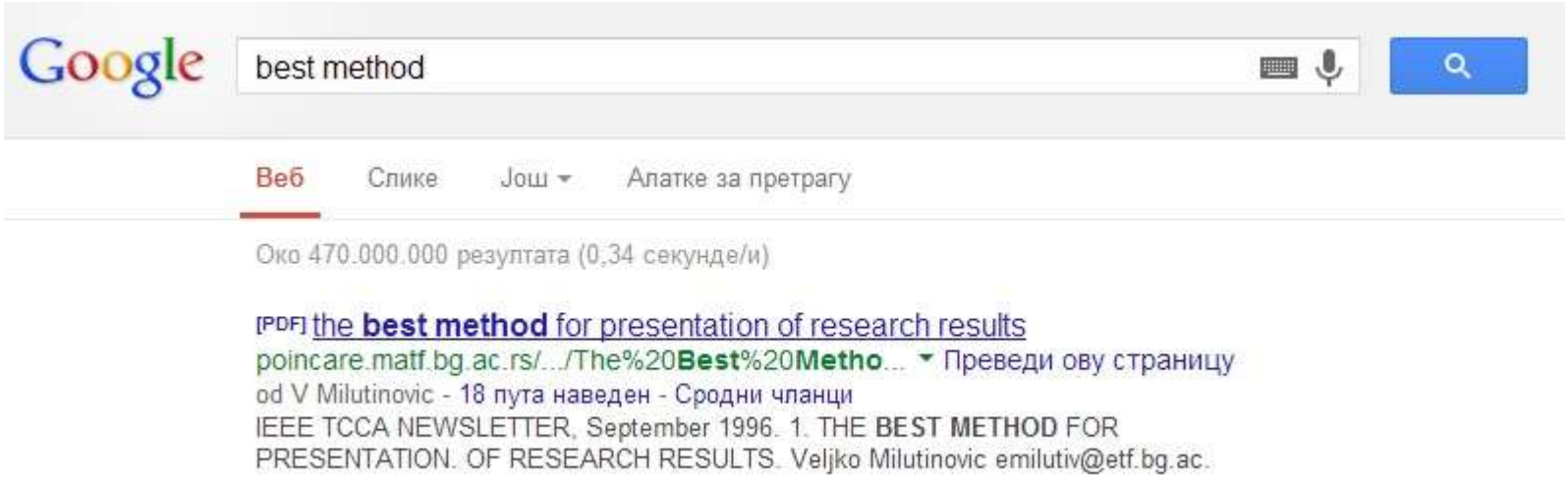


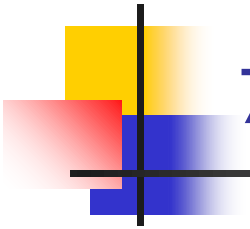
8: WRITING A RESEARCH/DEVELOPMENT PAPER

- Best method .doc
- Over one million hits at Google

8: WRITING A RESEARCH/DEVELOPMENT PAPER

- Best method .doc
- Over one million hits at Google



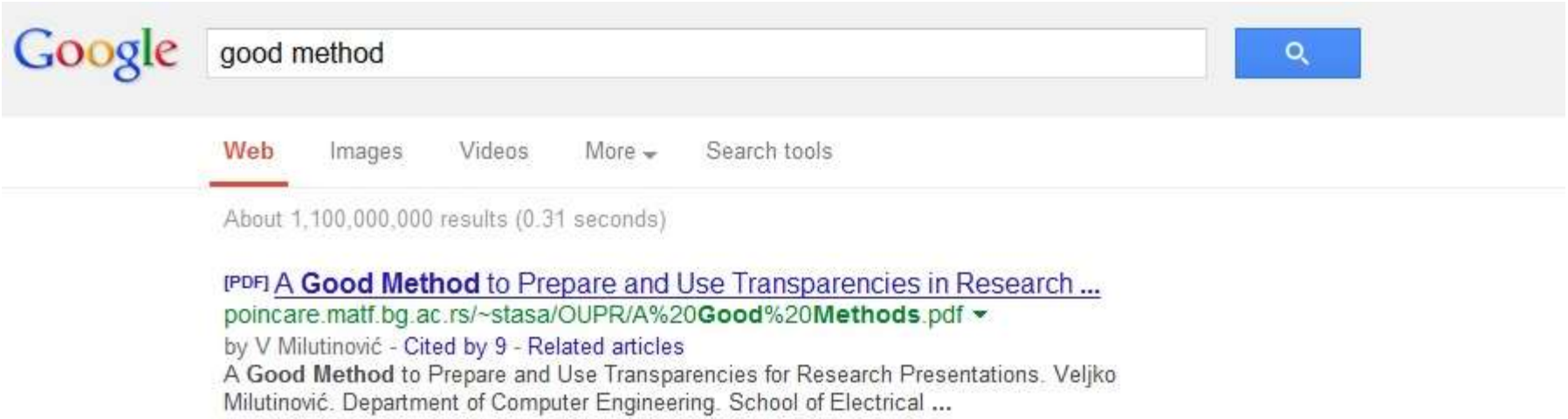


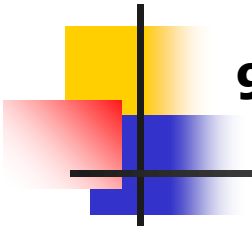
7: WRITING A SURVEY/OVERVIEW PAPER

- Good method .ppt
- Over one billion hits at Google

7: WRITING A SURVEY/OVERVIEW PAPER

- Good method .ppt
- Over one billion hits at Google

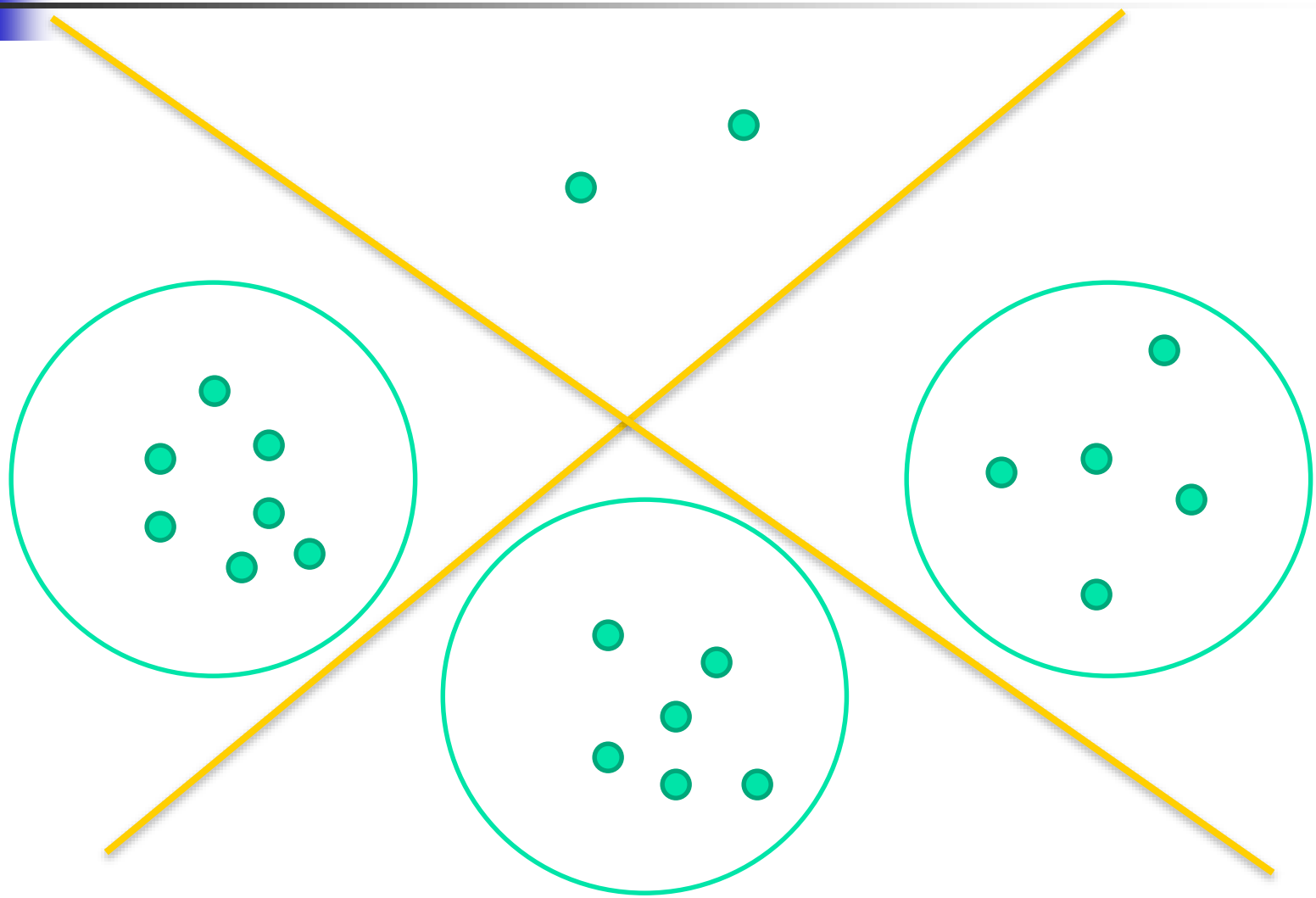




9! How to Make Your Own E-Commerce Site: Step by Step

- YAHOO! Store
  - The easiest way to make your business Internet-enabled, effective and efficient.
- ecBuilder
  - Software package, intended for more advanced eCommerce solutions, still very easy to use.
- Microsoft Site Server Commerce Edition
  - The most powerful tool, for the large and most advanced e-commerce solutions.

10: MindGenomics and LinearRegression



11: DataMining

Knowl Inf Syst (2008) 14:1–37
DOI 10.1007/s10115-007-0114-2

SURVEY PAPER

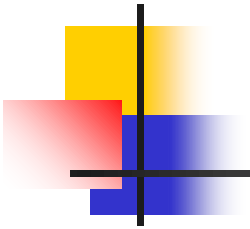
Top 10 algorithms in data mining

Xindong Wu · Vipin Kumar · J. Ross Quinlan · Joydeep Ghosh · Qiang Yang · Hiroshi Motoda · Geoffrey J. McLachlan · Angus Ng · Bing Liu · Philip S. Yu · Zhi-Hua Zhou · Michael Steinbach · David J. Hand · Dan Steinberg

Received: 9 July 2007 / Revised: 28 September 2007 / Accepted: 8 October 2007
Published online: 4 December 2007
© Springer-Verlag London Limited 2007

Abstract This paper presents the top 10 data mining algorithms identified by the IEEE International Conference on Data Mining (ICDM) in December 2006: C4.5, *k*-Means, SVM, Apriori, EM, PageRank, AdaBoost, *k*NN, Naive Bayes, and CART. These top 10 algorithms are among the most influential data mining algorithms in the research community. With each algorithm, we provide a description of the algorithm, discuss the impact of the algorithm, and review current and further research on the algorithm. These 10 algorithms cover classification,

12: Making a Brand



Ten Idea Generation Methods*

Mendeleyevization: Inductor versus Catalyst (M1 vs M2)

Definition:

If one of the classification class includes no examples,
it first has to be checked why is that so.
If it is so because it makes no sense, an appropriate explanation is in place.
If it is so because the technology or the applications are not yet ready
for such an approach, one can act in the same way as the famous chemists Mendeleyev:
Empty positions in any classification are potential avenues leading to new inventions.
We refer to such an approach as: Mendeleyevization.

A: Mendeleyevization (Inductor versus Catalyst) – M1 vs M2

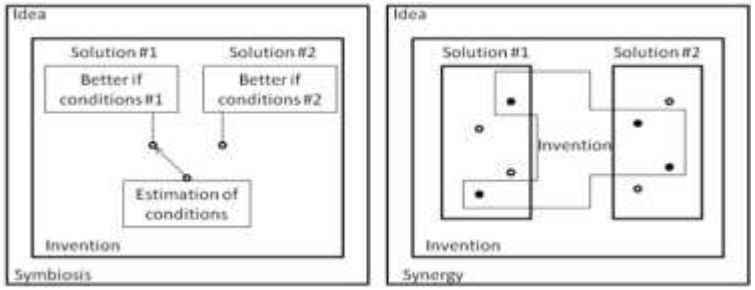
Examples:

As far as M1/M2, the famous classification of computer systems by Mike Flynn
(SISD, SIMD, MISD, MIMD) initially included no examples of the MISD type.
Later on, a DFT machine (generated using the M1 method)
was categorized as an MISD machine [Milutinovic86A],
as well as one pipelined machine (generated using the M2 method),
namely [Milutinovic87C];
the DFT served as an **inductor**, and pipeline as a **catalyst**.
Other popular examples are related to various signal processors and process accelerators.

Hybridization: Symbiosis versus Synergy (H1 vs H2)

Definition:

Sometimes two classification classes can be combined,
in order to obtain a hybrid solution (hybridization).
Hybrid solutions can be **symbiotic** (measuring the conditions in the environment
and switching from one approach to the other, so that each approach is active all the time,
while the conditions are such that it provides better performance compared to the other approach)
or **synergistic** (creating a new approach, which, for each particular solution element
takes the better solution element of two different approaches).



B: Hybridization (Symbiosis versus Synergy) – H1 vs H2

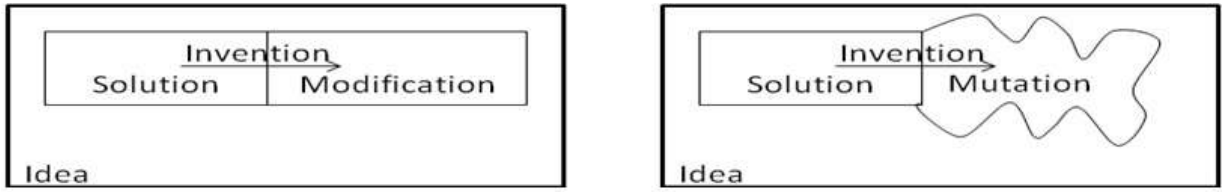
Examples:

As far as H1/H2, the essence of [Milutinovic85] is that two algorithms are combined into one
on the either-one-or-the-other basis (using the H1 method),
and on a combine-inherent-details basis (using the H2 method) in [Milutinovic87B].
Other popular examples include hybrid computers
or computers that use special purpose accelerators,
when appropriate data/process patterns are located.

Transdisciplinarization: Modifications versus Mutations (T1 vs T2)

Definition:

Often times, good new ideas get generated if algorithms, procedures,
ways of thinking, re ported from one field to another field,
along the lines of transdisciplinary research methodologies (transdisciplinarization).



C: Transdisciplinarization (Modification versus Mutation) – T1 vs T2

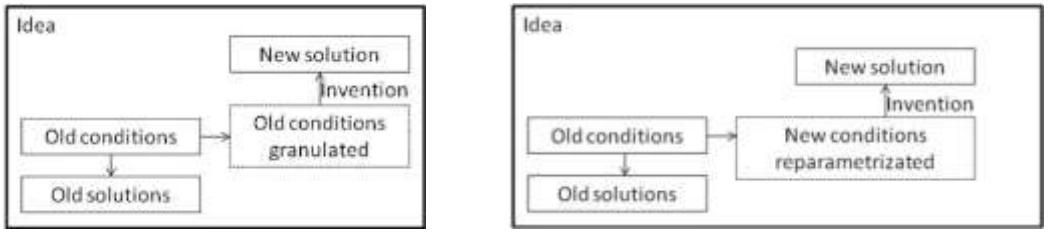
Examples:

As far as T1/T2, [Milutinovic86B] ports algorithms from Silicon to GaAs (using the T1 method),
and introduces appropriate modifications along the process,
while [Milutinovic87A] creates a proposal for a novel computer architecture
(using the T2 method),
along the analogies with a biological honeycomb.
Other popular examples include porting of the FFT
from seismic signal processing to speech signal processing
(**modification**),
or introduction of mathematical neural networks inspired by biological neural networks (**mutation**).

Remodelling:
Granularization versus Reparametrization (R1 vs R2)

Definition:

Sometimes it is simply the best to take a research direction different (even opposite)
compared to what others take (retrajectorization using remodeling).
The different (opposite) research direction makes sense either if a more detailed set of parameters is in place (**granularization**, due to model changes because of application changes),
or because parameters of the environment have changed permanently
(**reparametrization**, due to model changes because of technology changes).
The two alternatives are referred to as granularization and reparametrization.



D: Remodeling (Granularization versus Reparametrization) – R1 vs R2

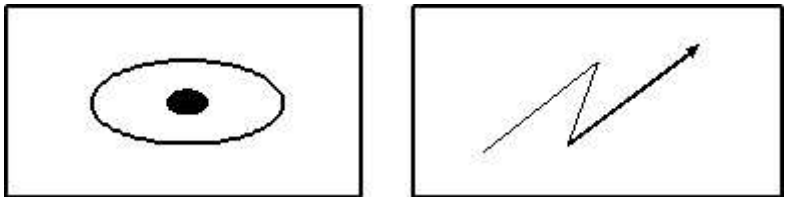
Examples:

As far as R1/R2, [Milutinovic88] offers a new algorithm (using the R1 method)
that makes sense if an environment is represented with a more detailed model,
while [Milutinovic89] offers a new solution in a changed environment (using the R2 method),
when a design has to be ported from Silicon to GaAs
(from the performance/price point of view, the best adder design for Silicon, the carry-lookahead adder,
is among the worst ones for GaAs, and the opposite: the worst adder for Silicon, the ripple-carry adder,
is among the best ones for GaAs; this is because GaAs gate delays depend on fan-out
and ripple-carry adders are characterized with only the minimal fan-out,
while the fan-out of the carry-lookahead adders depends on the word size, and can grow infinitely).
Other popular examples are related to concept modeling in AI based on graphical representation with icons
(in a model with fewer icons, one can make a conclusion which is different, and often times even opposite, compared to a conclusion made from a model with a much larger number of icons);
also, when the environment changes (for example, the ratio of processing speed to communication speed changes), a different type of supercomputing network becomes optimal.

Unorthodoxization: ViewFromAbove versus ViewFromInside (U1 vs U2)

Definition:

This category encompasses the approaches that are difficult to classify:
*Sometimes one sees something that others did not see for decades or centuries (**ViewFromAbove**)*
*or one gets stroked by an idea of a genius with no ground in existing research (**ViewFromInside**).*

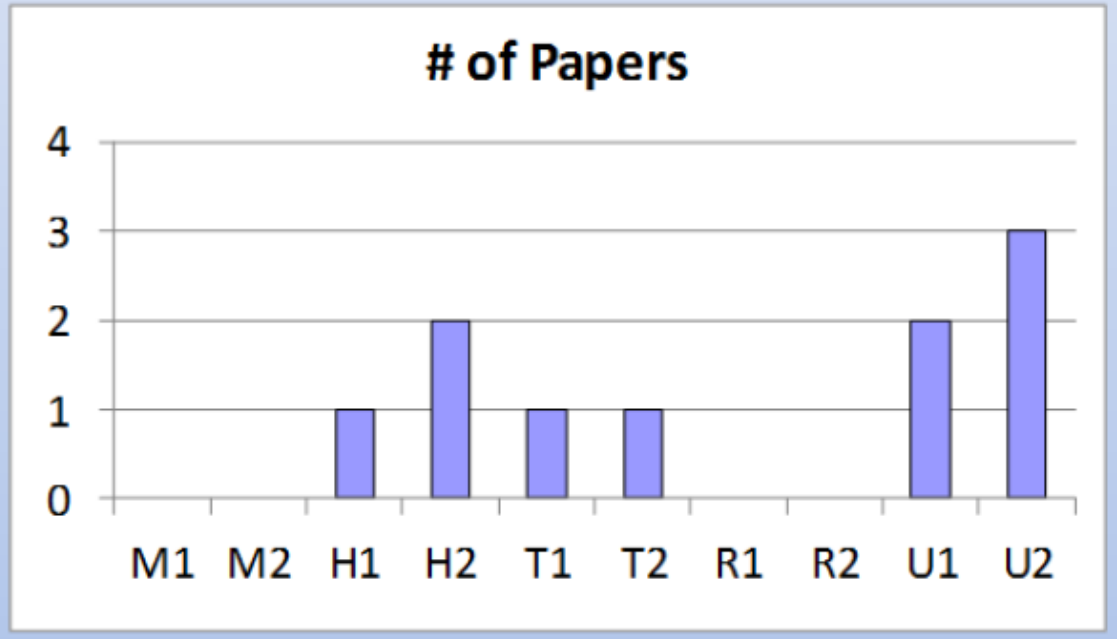


E: Unorthodoxization (ViewFromAbove versus ViewFromInside) – U1 vs U2

Examples:

As far as U1/U2, [Milutinovic2000] generated an innovation
after trying to make a holistic view (U1),
and [Milutinovic2001] introduces an idea after an effort is made
to understand the intrinsic details of the problem (U2).
Other popular examples include the contributions
of Nobel Laureates Martin Perl and Jerome Friedman.

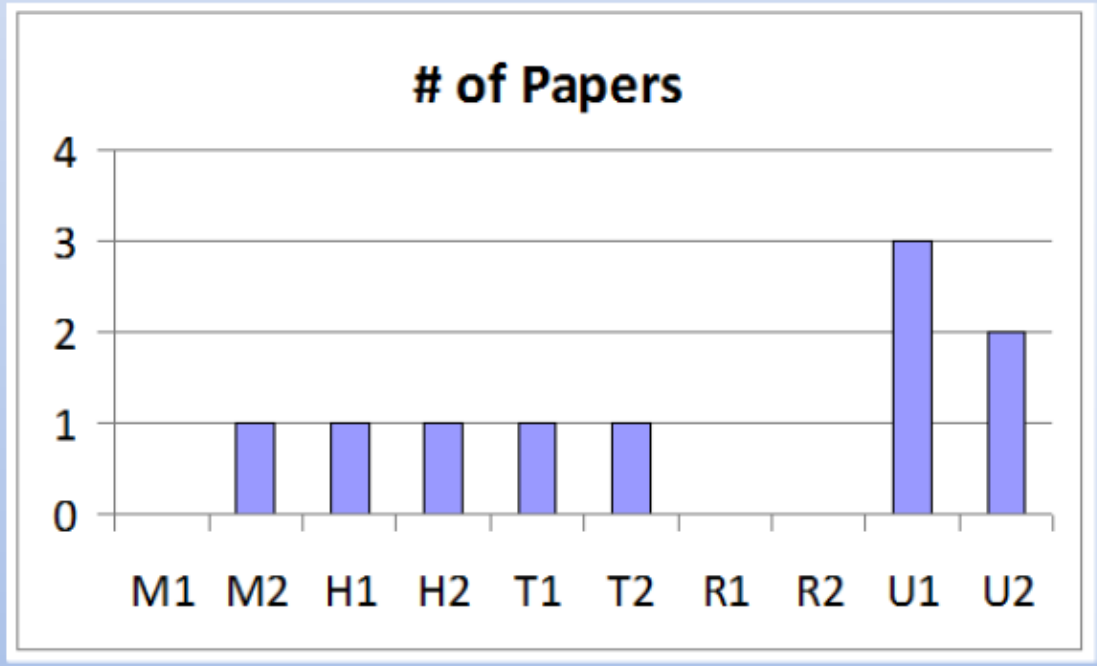
Examples from the Turing Award



Number of Turing Awards based on the given innovation method

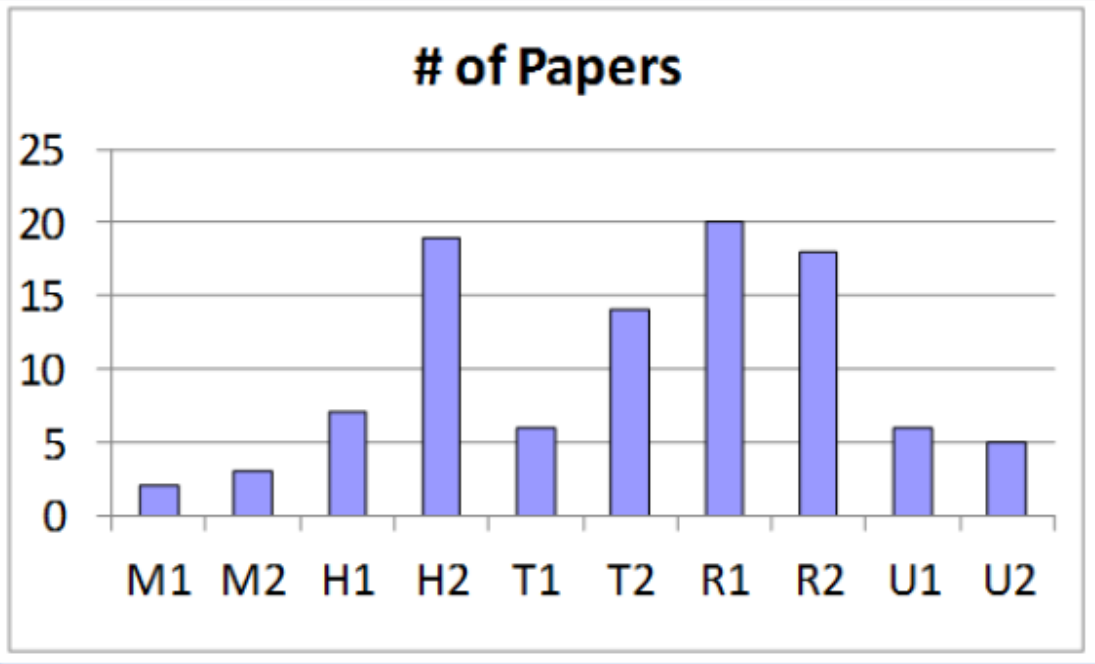
Alan J. Perlis (1966), Maurice V. Wilkes (1967), Richard Hamming (1968), Marvin Minsky (1969), James H. Wilkinson (1970), John McCarthy (1971), Edsger W. Dijkstra (1972), Charles W. Bachman (1973), Donald E. Knuth (1974), John Backus (1977)

Examples from the Nobel Laureate Research



K. Arrow,
L. Cooper,
P. DeGennes,
J. Friedman,
S. Glashow,
H. Kroto,
E. Maskin,
M. Perl,
B. Richardson,
K. Wilson

Examples from the List of Top 500 Computer Scientists of the World



Experiences with PhD Students of the Authors of this Research

Researcher	Research domain	Method
Drazen Draskovic	mutation algorithms for genetic search [Draskovic2012]	H1
Bojan Furlan	opinion mining for social networks [Furlan2011]	H1
Nemanja Kojic	data mining for wireless sensor networks [Kojic2012]	U1
Marko Misic	interconnection networks for multiprocessor systems [Misic2011]	R2
Milos Cvetanovic	system software for wireless sensor networks [Cvetanovic2008]	H1
Zaharije Radivojevic	application software for wireless sensor networks [Radivojevic2008]	H1
Zarko Stanisavljevic	computing infrastructure for distant education [Stanisavljevic2011]	H1
Zivojin Sustran	cache management for multiprocessor systems [Sustran2012]	T2
Djordje Djurdjevic	of computer graphics for mission applications [Djurdjevic2011]	R1
Sasa Stojanovic	hybrid computing for supercomputer architecture [Stojanovic2012]	H1

Classified References Used in the Educational Process

- **M1: Mendeleyevization/Inductor**
  [Milutinovic86a] *Milutinovic, V., Fortes, J., Jamieson, L.,A Multiprocessor Architecture for Real-Time Computation of a Class of DFT Algorithm,IEEE Transactions on Acoustics, Speech, and signal Processing, Aol. ASSP-34, No. 5, October 1986, pp. 1301-1309.(impact factor 1.463/1992).*

- **M2: Mendeleyevization/Catalyst**
  [Milutinovic87c] *Milutinovic, V.,A Simulation Study of the Vertical-Migration Microprocessor Architecture,IEEE Transactions on Software Engineering, Vol. SE-13, No. 12, December 1987, pp. 1265-1277.*

- **H1: Hybridization/Symbiosis**
  [Milutinovic85] *Milutinovic, V.,A Microprocessor-Oriented Algorithm for Adaptive Equalization,IEEE Transactions on Communications, Vol COM-33, No 6, June 1985, pp. 522-526.(impact factor 1.512/2010).*

- **H2: Hybridization/Synergy**
  [Milutinovic87b] *Milutinovic, V., Lopez-Benitez, N., Hwang, K.,A GaAs-Based Microprocessor Architecture for Real-Time Applications,IEEE Transactions on Computer, VolC-36, No 6, June 1987, pp. 714-727.(impact factor 1.822/2010).*

- **T1: Transdisciplinarization/Modification**
  [Milutinovic86b] *Milutinovic, V.,GaAs Microprocessor Technology,IEEE Computer, Vol 19, No. 10, October 1986 (Invited, Guest Editor's Introduction), pp. 10-15.(impact factor 2.205/2010).*

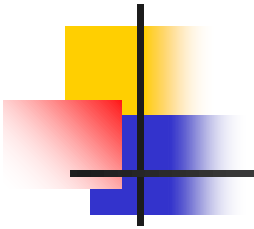
- **T2: Transdisciplinarization/Mutation**
  [Milutinovic87a] *Milutinovic, D., Milutinovic, V., Soucek, B.,The Honeycomb Architecture,IEEE Computer, Vol. 20, No. 4, April 1987 (Open Channel), pp. 81-83.(impact factor 2.205/2010).*

- **R1: Remodeling/Granularization**
  [Milutinovic88] *Milutinovic, V., A Comparison of Suboptimal Detection Algorithms Applied to the Additive Mix of Orthogonal Sinusoidal Signals, IEEE Transactions on Communicatiions, Vol. COM-36, No. 5, May 1988, pp. 538-543.*

- **R2: Remodeling /Reparametrization**
  [Milutinovic89] *Milutinovic, V., Bettinger, M., Helbig, W.,Multiplier/Shifter Design Trade-offs in a 32-bit Microprocessor,IEEE Transactions on Computers, Vol. 38, No. 6, June 1989, pp. 847-880.(impact factor 1.822/2010).*

- **U1: Unorthodoxization/ViewFromAbove [Milutinovic2000]** *Milutinovic, V., Cvetkovic, D., Mirkovic, J., "Genetic Search Based on Multiple Mutation Approaches," IEEE Computer, 2000. (impact factor 1.822/2010).*

- **U2: Unorthodoxization/ViewFromInside [Milutinovic2001]** *Milutinovic, V., Ngom, P., Stojmenovic, I., "STRIP --- A Strip Based Neural Network Growth Algorithm for Learning Multiple-Valued Functions," IEEE Transactions on Computers, 2001. (impact factor 1.822/2010).*

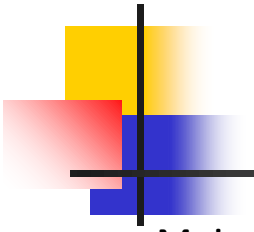


A Short Course for PhD Students in Science and Engineering: ”How to Write Papers for JCR Journals”

(A) survey papers

(B) research papers

Major Contributions of the Two Paper Types



Major contributions of the two paper types are as follows:

a) For a survey paper:

1) A novel classification of existing approaches to the problem, using a well thought set of classification criteria.
2) Presentation of each approach using the same template and the same type of figures, so an easy comparison is possible.
3) Some wisdom related to future research trends.

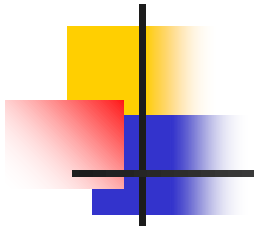
b) For a research paper:

1) Introduction of a new original idea.
2) Comparison of that idea with the best one from the open literature, using the previously built tools:

For how much is it better and under what conditions?

3) In addition to a performance oriented comparison, any research paper also has to include a complexity oriented comparison.

1. Survey Papers



Selection of the topic for a survey must satisfy

the following requirements:

1) The field is newly emerging.
2) Popularity of the field will grow over time.
3) A critical number of papers with new algorithms/approaches does exist (at least twenty to forty).
4) A survey paper does not exist.
5) The PhD student worked before in a related scientific field.
6) The PhD student is enthusiastic about the particular field of his/her tutorial paper.

Find papers using Google Scholar,

Citation Count for old papers and Locality Principle for new papers!

1. Survey Papers

- With the binary (or n-ary) criteria, one can create either a tree-like classification or a cube-like classification, as indicated in Figures 1 and 2 [Vukasinovic2012].

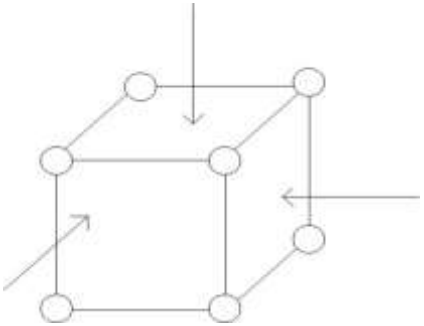


FIGURE 1. A tree-like classification:
Classes are only at the leaves of the tree.

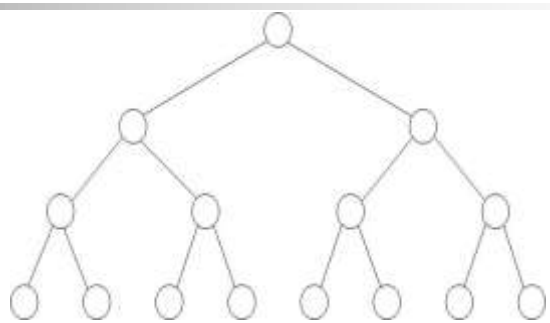


FIGURE 2. A cube-like classification:
Classes can exist also at points inside the cube, as pointed to by the three arrows.

- With a tree-like classification, one can classify only the approaches that entirely belong to a specific class.
  With a cube-like classification, one defines a space in which inner points include, to some extent, characteristics of all existing classes

- What is useful, is to prepare a figure which includes the following:
  - The classification criteria.
  - The classification
  - The technical mnemonics.
  - The symbolic mnemonics.
  - The number of selected examples per class.
  - The full list of references of selected examples.
  - The vector of relevant characteristics.

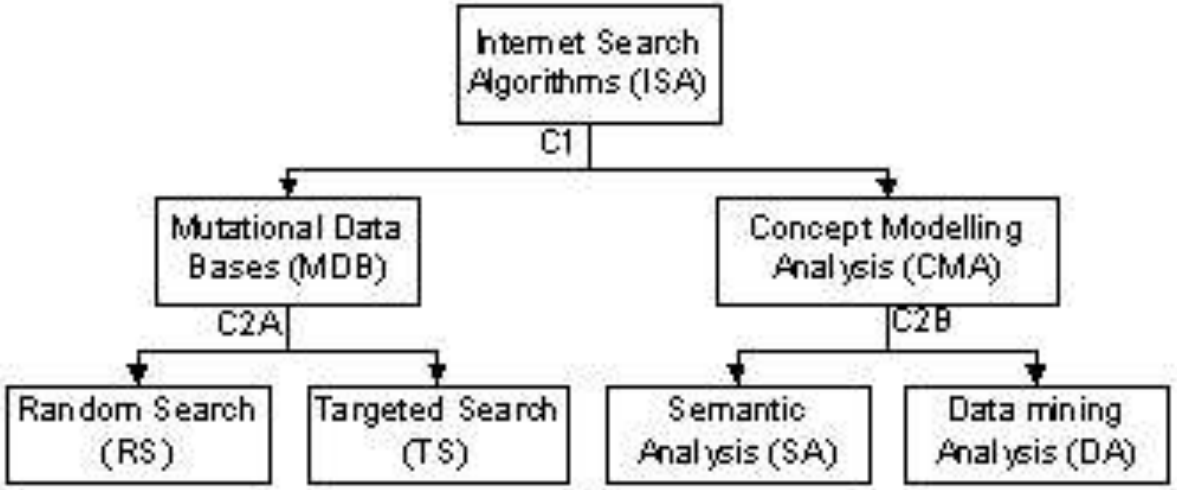


Figure 3. Classification of Internet Search Algorithms

Legend: C1 (criterion #1) = Retrieval-oriented vs Analysis-oriented
C2A (criterion #2, in the MDB path) = Random Search vs Targeted Search
C2B (criterion #2, in the CMA path) = Semantics-oriented vs Datamining-oriented

Technical Names	Random Search (RS or RS/MDB)	Targeted Search (TS or TS/MDB)	Semantic Analysis (SA or SA/CMA)	Data-mining Analysis (DA or DA/CMA)
Symbolic Names	Lion	Jaguar	Tiger	Panthera
Number of Surveyed Contributions	4	4	4	4
References	[Nikolic2011a] Nikolic, B., "Expert Systems," WUS Austria Educational Publishing and University of Belgrade, Classroom Textbook, June 2011.	[Milutinovic2000a] Milutinovic, V., Cvetkovic, D., Mirkovic, J., "Genetic Search Based on Multiple Mutation Approaches," IEEE Computer, November 2000, vol. 33, issue: 9, pp. 118-119.	[Nikolic2011b] Furlan, B., Sivacki, V., Jovanovic, D., Nikolic, B., "Comparable Evaluation of Contemporary Corpus-Based and Knowledge-Bases Semantic Similarity Measures of Short Text," JITA, vol. 1, no. 1, pp. 65-72, ISSN: 2232-962, June 2011.	[Milutinovic2000b] Milutinovic, V., Knezevic, P., Radunovic, B., Casselman, S., Schewel, J.,"Obelix Searches Internet Using Customer Data,"IEEE Computer, July 2000, vol. 33, issue: 7, pp. 104-107.
	[Nick2001] [ ] [ ]	[Simon2009] [Mirkovic1999] [Chen1997]	[Gordon2006] [Leroy2003] [Wang2006]	[Al-Dallal2009] [Hu2007] [Freitas2001]
-ability#1 ... -ability#N				

TABLE A: SUMMARY OF THE APPROACHES THAT LED TO THE CLASSIFICATION PROPOSED IN THIS PAPER

Survey Papers

When presenting each particular example, one can use the template presented next:

1) Seven Ws about the survey example (Who, What, When, Where, Why, for Whom, hoW) - 7.
2) Essence (it is extremely difficult to give entire essence in only one sentence) - k.
3) Structure (at this place, one can insert a call to a figure,
   like in [Draskovic2012]) - 3.
4) Some relevant infrastructure details - l.
5) Algorithm (refer to a figure) - 3.
6) Example (here one can call a figure that explains an example using a pseudo-code,
   like in [Draskovic2012];
   ideally, the same application case should be used for all surveyed examples) - m.
7) Seven As pros and cons + trends and ideas,
   as well as Author's opinion of this approach and its potentials
   in the domains of technology, architecture, and application - 7.

For short surveys, each template element is a sentence.

For long surveys, each template element is a paragraph.

For books, each template element can be a page, or more.

Parts of the Figure Caption

1. Title
2. Legend
3. Discussion
   (description + explanation + consequences)

Note: Symbols in the figure should be language independent.

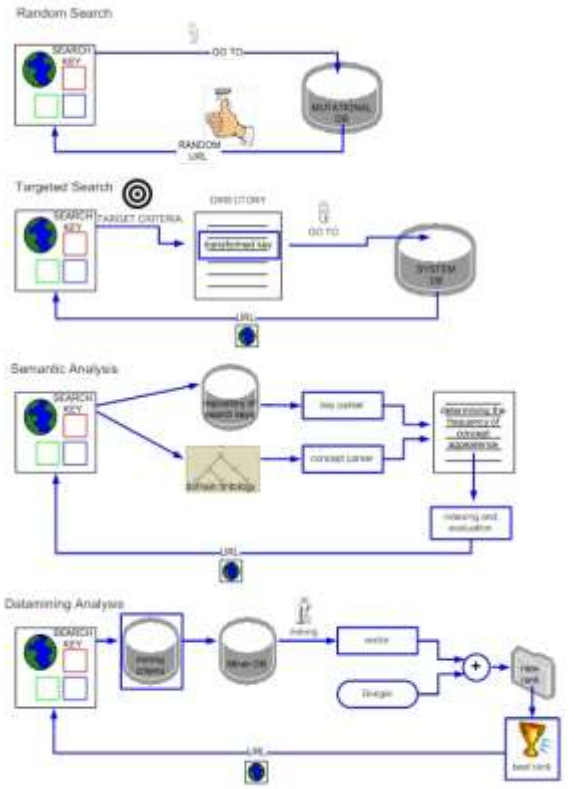


FIGURE 4: Generalized Structure of the Search Classes
Legend:
DB = Database;
URL = Type of URI that is used to describe
the location of a specific document;

```
PseudoCode
    Procedure#X;
    Procedure#Y;
    Procedure#Z;
    ...
EndPseudoCode
```

FIGURE 5: PseudoCode
that demonstrates behavior of an example,
in the case of a specific application;
it is advised that the same application is used
with all examples.

2. Research Papers

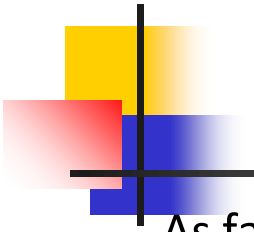
The major purpose of the research paper is to describe an innovation
and to demonstrate that, under certain conditions,
it has a better performance and/or complexity,
compared to the best one from the open literature.
The major steps in the process are:

1) To create an invention.

2) To perform a rigorous analysis,
to demonstrate that the invented solution is better
than the best one from the open literature under a specific set of conditions,
and to show what these conditions are
and for how much is it better.

3) To asses complexity and to write the paper using a methodologically correct template.

Research Papers



As far as the presentation of the research results, the students are told that each research paper should contain the following twelve sections:

Introduction

1) Problem statement
2) Existing solutions
3) The proposed solution
4) Details
5) Axioms, conditions, and assumptions of the analysis to follow
6) Mathematical analysis
7) Simulation analysis to show performance
8) Implementation analysis to show complexity
9) Conclusion
10) Acknowledgments
11) Annotated references

1. Introduction

The minimum introductory text

should contain the following three paragraphs:

a) About the general field of this research.

b) About the specific field of this research.

c) About the vision (viewpoint) of this research,
as well as the goal (battle),
and mission (war) of this research.

2. Problem Statement

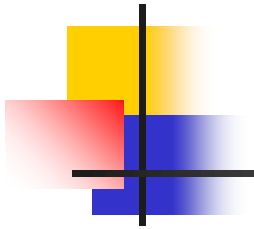
The following elements are obligatory:

a) Problem definition.

b) Why is the problem important.

c) Why will the importance of the problem grow over time.

3. Existing Solutions



Existing solutions and their drawbacks,
looking from the viewpoint defined in the introduction,
and having in mind the goals defined in the introduction.
Elements of this section are:

- A brief classification of the best solution from the open literature.
- Short description of each relevant solution.
- A detailed criticism of each presented solution,
  especially in domains in which the proposed solution
  is expected to be better.

4. The Proposed Solution

The proposed solution and its essence,

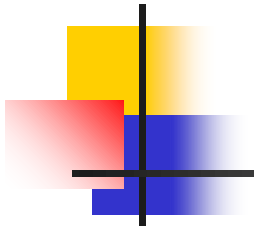
and why is it supposed to be better

compared to the best solution from the open literature;

elements of this section are:

a) Philosophical essence of the proposed solution.
b) Why the proposed solution
   is without drawbacks of existing solution(s).
c) What is the best methodology
   to prove the superiority of the proposed solution,
   and under what conditions that holds.

5. Details



This section should contain details of the best one among the existing approaches and of the proposed solution. The relevant details should be grouped into categories. For example:

a) Hardware details.

b) System software details

c) Application software details.

6. Axioms, Conditions, and Assumptions of the Analysis to Follow

a) Axioms refer to axiomatic standpoints.

b) Conditions refer to realistic specifiers of the environment.

c) Assumptions refer to simplifications that make the analysis easier, without jeopardizing on the quality of the final result.

7. Mathematical Analysis

a) Axioms, conditions, and assumptions are described mathematically.

b) Closed or open form formulae are derived for the major performance measures.

c) Closed or open form formulae are derived for the major complexity measures.

8. Simulation Analysis to Show Performance

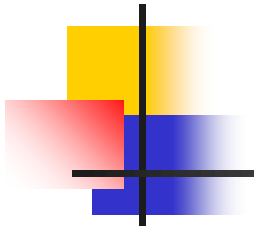
a) Simulator, logical structure

   and user interface are described.

b) Simulation experiments are described.

c) Simulation results are discussed.

9. Implementation Analysis to Show Complexity



a) Implementation strategy is discussed for the chosen technology.

b) Implementation details and complexity are presented.

c) If a prototype is implemented, show some characteristic measurement.
If a prototype is not implemented, give some implementation guidelines.

10. Conclusion

a) Summary of what was done
and to what extent are the initial goals achieved
(vision + goal + mission).

b) To whom is that of benefit.

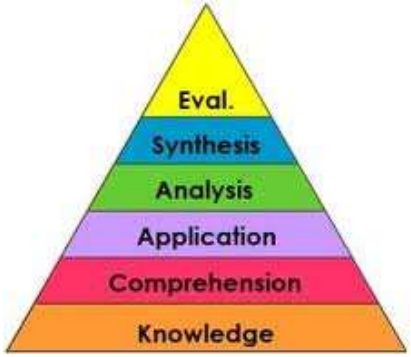
c) Newly open problem for further research.

PowerPoint Presentations: Condicios Since Qua Non

n/N (Expectations + Attention + TimeSavings)

5/7 (Tables + Plots + Figures + Graphs + Photos + Animations + Movies)

Semantic Breaks (One Line - One Thought)



Semantic Breaks: An Example (Bad)

Research Issues of Importance for Distributed Shared Memory

Hardware Issues (those to be fully implemented in hardware)

Software Issues (those to be fully implemented in software)

Hybrid Issues (those to be partially implemented in hardware and partially in software)

Semantic Breaks: An Example (Good)

Research Issues of Importance

for Distributed Shared Memory

Hardware Issues

(those to be fully implemented in hardware)

Software Issues

(those to be fully implemented in software)

Hybrid Issues

(those to be partially implemented in hardware and partially in software)

Priestess of Delphi: Pythia

Ibis, redibis nunquam,
in belo peribis

Ibis, redibis,
nunquam in belo peribis

Romeo and Juliet: Prologue

Two households, both alike in dignity,
In fair Verona, where we lay our scene,
From ancient grudge break to new mutiny,
Where civil blood makes civil hands unclean.
From forth the fatal loins of these two foes
A pair of star-cross'd lovers take their life;
Whose misadventured piteous overthrows
Do with their death bury their parents' strife.
The fearful passage of their death-mark'd love,
And the continuance of their parents' rage,
Which, but their children's end, nought could remove,
Is now the two hours' traffic of our stage;
The which if you with patient ears attend,
What here shall miss, our toil shall strive to mend.

If properly used, decreases the probability that people laugh about us when we make mistakes (e.g., when the noun and the related adjective are not next to each other).

What is Better?

With semantic breaks:

```
function DayOfWeek(day, month, year) {
    var a = Math.floor((14 - month) / 12);
    var y = year - a;
    var m = month + 12 * a - 2;
    var d = (day + y + Math.floor(y / 4) - Math.floor(y / 100) +
            Math.floor(y / 400) + Math.floor((31 * m) / 12))  % 7;
    return d;
}
```

Essence: The higher the level of abstraction in the text,
the more benefit from semantic breaks

Also: The more tired the audience,
the more benefits from semantic breaks

What is Better?

With semantic breaks:

```
function DayOfWeek(day, month, year) {
    var a = Math.floor((14 - month) / 12);
    var y = year - a;
    var m = month + 12 * a - 2;
    var d = (day + y + Math.floor(y / 4) - Math.floor(y / 100) +
            Math.floor(y / 400) + Math.floor((31 * m) / 12))  % 7;
    return d;
}
```

Without semantic breaks:

```
function DayOfWeek(day, month, year) { var a = Math.floor((14 - month) / 12);
var y = year - a; var m = month + 12 * a - 2; var d = (day + y + Math.floor(y /
4) - Math.floor(y / 100) + Math.floor(y / 400) + Math.floor((31 * m) / 12))  % 7
return d;}
```

Question: How long it takes to figure out the essence of the above?
Answer: BioPhysics + PsychologyAnalytics!

Q&A

vm@etf.rs