Towards an ISO/IEC 15504-Based Process Capability Profile Methodology for Process Improvement (PRO2PI)

Clenio F. Salviano
CenPRA MCT
Campinas, SP – Brazil
clenio.salviano@cenpra.gov.br

Mario Jino
DCA FEEC UNICAMP
Campinas, SP – Brazil
jino@dca.fee.unicamp.br

Manuel de Jesus Mendes
UNISANTOS
Santos, SP – Brazil
mj.mendes@terra.com.br

Abstract

The ISO/IEC 15504’s Continuous Architecture offers an opportunity to advance the benefits of process improvement, over the CMM-based staged architecture. It allows the definition of a specific hierarchy of Process Capability Profiles (PCP) that better addresses the organization business context to be used as reference for an effective improvement. A PCP is composed by a set of processes, each one in a capability level. However, as a framework for process assessment, not process improvement, 15504 does not provide adequate support for defining PCPs, making difficult to use this opportunity. A research effort towards the development and experimentation of a methodology for the definition, operation and utilization of a hierarchy of specific and dynamic ISO/IEC 15504-Based Process Capability Profile for Process Improvement (PRO2PI) is under way. This article describes the rationality for this research, the main elements of the PRO2PI Methodology and some experiences and examples with its earlier versions.

1. Introduction

Software Process Improvement (SPI) has given a new breathe in the software engineering struggle to improve quality in software. SPI moved the focus away from a frequently changing technology to a more stable and feasible process management view. Such a stable view is based on the capability of the processes used by an organization to perform the acquisition, supply, development, operation, evolution, and support of software systems. SPI has grown up during the 80s to address the increasing complexity and criticality of software development activities. The success of SPI based on the Capability Maturity Model for Software (SW-CMM) [9] is responsible for the dissemination of SPI. SW-CMM organized the software engineering practices into four fixed cumulative maturity levels to guide the evolution of organizational processes for software development projects based on requirement agreement. SW-CMM defined the one dimensional staged architecture.

The new ISO/IEC 15504 International Standard (IS) for Process Assessment [5] defines a framework composed by six sequential process capability levels, and by requirements for process assessment process, process reference model and process assessment model. It also defines an exemplar process assessment model for software (15504-5). The previous 1998 version of 15504 was published as an ISO/IEC Technical Report (TR) [4]. The TR version was specific for software processes, while the new IS version may be applied to any human intensive process. 15504 introduced the two dimensional continuous architecture, composed by process and process capability. Using a continuous model, an organization selects a set of processes to guide an assessment for continuous improvement or capability determination.

CMMI [2], the new framework from SEI, defines models with two representations: one as a staged model, as the SW-CMM, and another with a continuous architecture similar to 15504.

This article introduces a research effort based on the view that the 15504’s Continuous Architecture offers an opportunity to advance the benefits of process improvement, over the CMM-based Staged Architecture. 15504 allows the selection of processes and capability level to be reached which best meets the organization’s business context, restrictions and objectives. The CMM-based staged models, imposes a predefined and generic improvement path.

Section 2 of this article presents the rationality and overview of PRO2PI methodology. Section 3 presents its major elements and Section 4 presents experiences and examples of PRO2PI.
2. Rationality and Overview

A continuous architecture separates best practices into two categories. The first category encompasses the practices related to “what we do”. These practices are organized into specific processes. The second category is related to “how well we do whatever we do”. These practices are organized into generic process capability levels. A Process Capability Profile (PCP) is a specific combination of processes and capability levels, which can be used as a reference for improvement. Figure 1 shows an example of a PCP.

The current debate about staged and continuous models associates staged model with proven path for organizational maturity and continuous models with improvement of individual processes [2,3]. In our view, however, a continuous model may, and should, also be used for organizational maturity. Although a continuous model is structured process by process, a specific and dynamic set of PCPs, organized as a hierarchy of “maturity levels” should be defined to guide an organizational improvement. In this way a staged model, such as SW-CMM, is a (good) example on how to use a continuous model. In this sense, a continuous architecture is an evolution of the staged one.

A PCP is the key element to make a bridge from a more generic continuous model to a more specific and dynamic staged model.

![Figure 1: Example of Process Capability Profile](image)

In our view, the models and frameworks for process capability have already had three generations, each one an evolution of the previous one in terms of better stability and increasing flexibility. Table 1 describes these three generations.

<table>
<thead>
<tr>
<th>Generation</th>
<th>Main Model or Framework, and year of release</th>
<th>Other Models</th>
<th>Architecture</th>
<th>Major Fixed Elements (Stability)</th>
<th>Major Variable Elements (Flexibility)</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>1st</td>
<td>SW-CMM v1.1 Model :1993</td>
<td>ISO/IEC TR 15504 Framework :1998</td>
<td>Continuous</td>
<td>Maturity Levels</td>
<td>Interpretation of the Maturity Levels</td>
<td>low flexibility, “one size fits all”, but ready to be used, responsible for dissemination of SPI</td>
</tr>
<tr>
<td>3rd</td>
<td>Capability Levels</td>
<td>Capability Levels</td>
<td>Capability Levels</td>
<td>Processes, Processes Profiles and their interpretation</td>
<td>better stability and flexibility balance, not ready to be used, needs methodology to be used</td>
<td></td>
</tr>
</tbody>
</table>

The key lesson learned from the success of SW-CMM is that a good hierarchy of PCPs, in that case represented by each maturity level, is very useful to guide the necessary and feasible improvement. The current limitation is that those PCPs are fixed.

In general, an organization should use a kind of staged model, with a hierarchy of PCPs, as a reference for improvement. As different organizations will need different PCPs to address different business contexts, or even, the same organization will need different PCPs at different times, there is a need for a methodology to support the definition, operation and usage of a hierarchy of dynamic and specific PCPs.

The objective of ISO/IEC 15504-based Process Capability Profile Methodology for Process Improvement, called PRO2PI¹, is to extend 15504 to provide such methodology using best features of continuous and staged architectures. As 15504 is a process assessment framework, it does not provide such support for process improvement. Using PRO2PI, the PCPs are organized into a capability hierarchy such as the maturity levels of staged models.

¹ The name PRO2PI (say pro-to-pi) uses the number “2” to mean the double PRO (Process Profile) and “to” Process Improvement (PI).
This article reflects the current version of PRO2PI. Figure 2 gives a high level view on PRO2PI and how to apply it.

3. PRO2PI Methodology

Based on the concept of PCP, defined in the previous section, the PRO2PI methodology can be briefly described using five major aspects: a notation to represent a PCP, properties of PCPs, operations, the relationships between PCPs and guidelines to define, operate and use PCPs. The current version of these aspects is described in the next sub-sections.

3.1. Representation

A PCP is represented using the following notation:

```
{"<Proc1d1>"..., "<Proc1d2>"..., "<Proc1d1>"..."
<CL<CLnumber>","<CL<CLnumber>"+"
<CL<CLnumber>"}
```

where `<ProcId>` is a process unique identification and `<CLNumber>` is the number of a 15504 capability level.

When many processes have the same capability level in a PCP, the following notation may also be used:

```
{"<ProcId1>"..., "<ProcId2>"..."}"+"CL<CLnumber>
```

3.2. Properties

To be useful and effective for process improvement, a PCP should possess, to a sufficient extent, at least five properties: a PCP should be relevance, attainable, systemic, traceable and dynamic.

Relevance: A PCP should represent a relevant state, clearly aligned with the organization business goals, strategy and context. From an analysis of the organization’s needs and existing stimuli for improvement, the improvement’s objectives are identified. These objectives are described in terms of quality, time to market, cost and customer satisfaction, and business value with information services, along with predictability and control of the delivery of information services and related risks [4]. All of them direct the definition of relevant PCPs for process improvement. Many approaches may be used for that, including Balanced Scorecard Framework and the Strategy Focus Organization [6] and the Goal-Problem Approach to SPI [10].

For example, an organization that needs to ensure the delivery of software products without major defects, may include a process for software testing at capability level 4,
all other software engineering processes at level 2 and a requirement elicitation process at level 3.

**Attainable:** The amount of effort and resources necessary to achieve the objectives represented by a PCP should be feasible. As all processes and capability level have similar granularity, the improvement effort could be estimated assigning one improvement point to each process and each capability level included in an improvement cycle. The number of improvement points will be a measure of the improvement effort\(^2\).

**Systemic:** Enough processes and capability levels should be included in a PCP to make it a system. As a system it should be as complete as necessary to provide an organizational capability, enough to provide results that are institutionalized and to be a plateau for a next level of improvement. It can be small, as for example, software construction process at level 1, or large, as an equivalent of CMMI maturity level 3. A PCP achieves the systemic property, when: it covers at least a “small life cycle” without “holes”, each process is at a capability level (not part of capability level, as for example, just a process attribute), and there is a compatible relationship among the capability levels of all processes.

Given that a set of processes covers best practices that are systemic at capability level 1, any capability level beyond that will also be systemic.

There are relationships between aspects of capability levels and specific processes. For example, to reach level 2, each execution of a process needs to be managed. Thus, a project management process may be used for this need. When a process is set to be at level 2, however, there is no need to also include project management process, because the process attribute already included the management aspects. Conversely, if there is an additional value in having a complete project management, then the project management process at an adequate capability level should also be included in the PCP. The capability levels provide the prerequisite aspects for the evolution of each process.

**Traceable:** Given the consolidation on the best generic practices for software, traceability between a PCP and relevant models is important for an organization. Available mappings between relevant models, such as, for example, ISO 9001, SW-CMM, 15504-5, CMMI and PMBoK, are useful for this traceability.

**Dynamic:** Usually it is not easy to determine the right PCP at the start of a SPI program and changes on the business context may occur during the SPI. Therefore the ability to adjust a PCP or a PCP hierarchy as needed is very important. Thus, there is a need to operate a PCP to incrementally reduce or increase its entire capability. In this way it is possible to have small steps for continuous improvements.

Note that there are two major forces to be balanced. The relevance property leads to a large PCP, while the attainability property leads to a small PCP.

### 3.3. Operations

There are five basic operations on PCPs to support the dynamic property:

- **Creation:** $PCPa_1 = \text{CreatePCP};$
- **Vertical increase:** $PCPa_2 = PCPa_1 + CLdelta;$
- **Vertical decrease:** $PCPa_2 = PCPa_1 - CLdelta;$
- **Horizontal composition:** $PCPa_3 = PCPa_1 + PCPa_2;$
- **Horizontal decomposition:** $PCPa_3 = PCPa_1 - PCPa_2;$

These operations allow the manipulation of PCPs to make them more appropriate to specific conditions. For example, if a specific PCPa1 is used as a reference for an improvement, and an analysis of an assessment results indicates a big gap, a vertical decrease of PCPa1 may be used to lower the reference to make the improvement more feasible. The original PCP may than be used as reference for a second improvement cycle.

### 3.4. Relationships

The most important relationship between two PCPs is the capability hierarchy. A PCPa1 has a capability higher than or equal to PCPa2 when, for each process in PCPa2 there is the same process in PCPa1, and the capability level for this process in PCPa1 is higher than or equal to the capability level in PCPa2. A hierarchy of PCPs in a cumulative sequence is a path for a sequential improvement, where each PCP is at the same time an improvement destiny and a plateau for the improvement based on the next PCP. A set like this is what we call a staged model, where each PCP is represented by the delta from the previous PCP and each delta is named “maturity level”.

### 3.5. Guidance

As guidance, we have been exploring three approaches:

- a) development of software patterns to help the understanding of process capability concepts, including PCP \([15,16]\);
- b) an experimental method for the definition of PCP \([17]\); and
- c) informal ways to define PCP (see example in \([19]\)).

---

\(^2\) This improvement measure is based on a suggestion from an anonymous reviewer.
A method was developed and applied for the first time in 1999 to support the definition of PCPs in SPI projects using continuous models, [18] (Figure 3).

![Example of results of act. 2 for an organization](image)

**Figure 3: Activities of EM-PRO2PI method**

This method is now named EM-PRO2PI: Experimental Method for PRO2PI Methodology. EM-PRO2PI consists of activities for gathering objective information which is used as suggestions for a subjective and collective decision. These activities are organized into four basic sequential phases: preparation, suggestions, definition and results, as illustrated in Figure 3. The suggestion phase consists of the four activities, through which objective suggestions for the PCP are gathered from:

a. the organization business objectives (activity 1);
b. importance and risk of each process for the organization (activity 2);
c. opinions about key business and technical issues (activity 3); and
d. experiences from SPI in other organizations and maturity levels from staged models (activity 4).

A two-dimension table is used to identify the importance and risk of each process. Each dimension has three values: low, medium and high. One dimension is on the importance of a process to the organization. The other dimension is on the risk incurred by the organization if the process continues to be performed as it is now (See example in Figure 3).

The definition phase consists of four activities. In the first one, the suggestions from the previous phase are mapped into processes and capability levels. A project management process, for example, may be mapped from a business goal related to a better control of projects. The other three activities are: the PCP is defined based on this mapping; and the PCP is revised and approved by the SPI sponsor.

4. Examples and Experiences

As indicated in Figure 2, PRO2PI may be used to:

a) create specific models,
b) define dynamic specific profiles,
c) improve processes, and
d) update a single or a hierarchy of PCPs.

This section presents examples and experiences with previous versions of what is now PRO2PI. Table 2 lists the processes used in these examples and experiences. The processes are taken from two models: ISO/IEC TR 15504-5 and SW-CMM. The processes from SW-CMM are key process areas interpreted as processes. Each process is uniquely identified with a "name id".

<table>
<thead>
<tr>
<th>name id</th>
<th>process name (from ISO/IEC TR 15504-5)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Aqui</td>
<td>Acquisition</td>
</tr>
<tr>
<td>AquPrep</td>
<td>Acquisition Preparation</td>
</tr>
<tr>
<td>CusAcc</td>
<td>Customer Acceptance</td>
</tr>
<tr>
<td>Supply</td>
<td>Supply</td>
</tr>
<tr>
<td>ReqEli</td>
<td>Requirements Elicitation</td>
</tr>
<tr>
<td>CusSup</td>
<td>Customer Support</td>
</tr>
<tr>
<td>SwReq</td>
<td>Software Requirements</td>
</tr>
<tr>
<td>SwCons</td>
<td>Software Construction (*1)</td>
</tr>
<tr>
<td>SwTest</td>
<td>Software Testing</td>
</tr>
<tr>
<td>ProjMan</td>
<td>Project Management</td>
</tr>
<tr>
<td>OrgAlign</td>
<td>Organization Alignment</td>
</tr>
<tr>
<td>ProcEst</td>
<td>Process Establishment</td>
</tr>
<tr>
<td>HRMan</td>
<td>Human Resource Management</td>
</tr>
<tr>
<td>Infrastr</td>
<td>Infrastructure (*2)</td>
</tr>
<tr>
<td>Measure</td>
<td>Measurement</td>
</tr>
<tr>
<td>Doc</td>
<td>Documentation</td>
</tr>
<tr>
<td>QuaAssur</td>
<td>Quality Assurance (*1): includes design, construction and integration (*2): adapted to software tools infrastructure</td>
</tr>
</tbody>
</table>

**Table 2: Processes used in Examples and Experiences**
Table 3 presents PCPs defined for six SPI projects, from 1999 to 2002. The table contains a unique identification for the organizational unit (OU), the year the PCP was defined, a brief characterization of the OU, the source model or models for the processes, the method used for the definition and the PCP defined.

Table 3: PCPs for six SPI projects

<table>
<thead>
<tr>
<th>d_year</th>
<th>OU type</th>
<th>Model(s)</th>
<th>Method</th>
<th>PCP</th>
</tr>
</thead>
<tbody>
<tr>
<td>OU.1:1999</td>
<td>Private, software development, product oriented, medium size</td>
<td>ISO/IEC TR 15504-5</td>
<td>EM-PRO2PI</td>
<td>[CurSup; ProjMan; OrgAlign; ProcEst; QuaAssur]:CL2</td>
</tr>
<tr>
<td>OU.2:1999</td>
<td>Private, software development, project oriented, small size</td>
<td>ISO/IEC TR 15504-5</td>
<td>EM-PRO2PI</td>
<td>[Supply; ProjMan]:CL2</td>
</tr>
<tr>
<td>OU.3:2000</td>
<td>Public, software specification and acquisition</td>
<td>ISO/IEC TR 15504-5 and SW-CMM</td>
<td>from SW-CMM 2</td>
<td>[RM; SSM; AquiPrep; CusAcc; ProjMan]:CL2</td>
</tr>
<tr>
<td>OU.4:2001</td>
<td>Public, software development</td>
<td>ISO/IEC TR 15504-5</td>
<td>EM-PRO2PI</td>
<td>[SwCons; Aqui; Doc; HRMan]:CL2</td>
</tr>
<tr>
<td>OU.5:2001</td>
<td>Private, software development, project oriented, small size</td>
<td>ISO/IEC TR 15504-5</td>
<td>Informal</td>
<td>[Supply; ReqEli; SwTest; ProjMan; Measure]:CL2</td>
</tr>
<tr>
<td>OU.6:2003</td>
<td>Military, software development, project oriented</td>
<td>ISO/IEC TR 15504-5 and SW-CMM</td>
<td>from SW-CMM 2</td>
<td>[RM; SSM; SwTest; Infrastr]:CL2</td>
</tr>
</tbody>
</table>

The SPI project for OU.1, OU.2 and OU.4 used the EM-PRO2PI method and the TR 15504-5 model.

OU.1 started the SPI actions based on the PCP defined in 1999 [18]. During the project the focus and the improvement plan were adjusted to include ISO 9001 as an additional reference. In 2002, a new process assessment was performed in OU.1 to capture the results from the improvement. At that time an expanded PCP was a better reference for the improvement: [CurSup;CL3; [ProjMan; SwReq; OrgAlign; ProcEst; Measure; QuaAssur]:CL2] [7].

In OU.3 the SPI project was requested having SW-CMM level 2 as the PCP. After a review of the OU context, the PCP was adjusted, because the OU did not have software implementation. Rather the OU focus was on understanding and anticipating client’s needs, defining the software and acquiring all software implementation. Therefore, a new PCP was defined to reflect that situation.

In OU.5, an informal, but careful, approach was used to define the PCP [19]. During the SPI project, more processes were improved, including the ones from engineering. The next step is to analyze the current situation, represent it as a PCP, and decide the new PCP for the next improvement cycle.

The OU.6 case can be explained using the application of some elements of what is now this version of PRO2PI. Table 4 shows four versions of the PCP. They were changed based on decisions and results from the project. It started with SW-CMM level 2 PCP and ended up with a different PCP.

Table 4: Operations on PCPs

<table>
<thead>
<tr>
<th>Step</th>
<th>Operation</th>
<th>Result PCP</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>P1 = Create</td>
<td>[RM, SPP, SPTO, SSM, SQA, SCM]:CL2</td>
<td>based on SW-CMM level 2</td>
</tr>
<tr>
<td>2</td>
<td>P2 = P1 + SwTest:CL2</td>
<td>[RM, SPP, SPTO, SSM, SQA, SCM, SwTest]:CL2</td>
<td>add Software Test, reference for an assessment</td>
</tr>
<tr>
<td>3</td>
<td>P3 = P2 – {SPP, SPTO, SSM, SQA}</td>
<td>[RM, SCM, SwTest]:CL2</td>
<td>after assessment, reduce scope to be feasible</td>
</tr>
<tr>
<td>4</td>
<td>P4 = P3 + Infrastr:CL2</td>
<td>[RM, SCM, SwTest, Infrastr]:CL2</td>
<td>add Infrastructure for software tools</td>
</tr>
</tbody>
</table>

As a result, for this project, the SW-CMM level 2 was increased and decomposed into two levels (see Table 5).

Table 5: Two levels for SW-CMM level 2

<table>
<thead>
<tr>
<th>PCP</th>
<th>Processes and capability level included</th>
</tr>
</thead>
<tbody>
<tr>
<td>PCP2.2</td>
<td>{SPP, SPTO, SSM, SQA}:CL2</td>
</tr>
<tr>
<td>PCP2.1</td>
<td>{RM, SCM, SwTest, Infrastr}:CL2</td>
</tr>
</tbody>
</table>

During the execution of the SPI projects from OU.1, OU.5 and OU.6, adjustments to the PCP were performed on an informal base. These experiences have been used to consolidate the PRO2PI methodology.
Tables 6 and 7 show two examples of specific PCPs hierarchies, similar to staged models. When the name of the process is in bold type, it means that this process was either included or its capability was increased in that PCP.

An organization reported the following strategy for improvement [20]. First they created and trained a group to perform system test and applied those tests in all products, getting evidence about the poor quality. Second, the activities on testing triggered an improvement in requirement specification. Third, they improved the customer support. Fourth, from the two extremes (requirements and test), the intermediate software engineering processes (design, construction and integration) were improved. Mapping this strategy into PRO2PI, we can see a five levels hierarchy of PCPs, as described in Table 6, applicable to this scenario.

### Table 6: A specific hierarchy of PCPs

<table>
<thead>
<tr>
<th>PCP</th>
<th>Processes and capability level in each PCP</th>
</tr>
</thead>
<tbody>
<tr>
<td>PCP_a</td>
<td>SwReq :CL1, SwCons :CL2, SwTest :CL2, CusSup :CL2</td>
</tr>
<tr>
<td>PCP_b</td>
<td>SwReq :CL2, SwCons :CL1, SwTest :CL2, CusSup :CL1</td>
</tr>
<tr>
<td>PCP_c</td>
<td>SwReq :CL2, SwCons :CL1, SwTest :CL2, CusSup :CL1</td>
</tr>
<tr>
<td>PCP_d</td>
<td>SwReq :CL1, SwCons :CL1, SwTest :CL1, CusSup :CL1</td>
</tr>
</tbody>
</table>

Another utilization of PRO2PI is in a project to create a staged model for micro and small software development organizations, with low maturity levels, through a hierarchy of small improvement steps. Table 7 shows a first version for the lowest six cumulative PCPs.

### Table 7: Another specific hierarchy of PCPs

<table>
<thead>
<tr>
<th>PCP</th>
<th>Processes and capability level in each PCP</th>
</tr>
</thead>
<tbody>
<tr>
<td>PCP_a</td>
<td>SwReq :CL1, SwCons :CL2, SwTest :CL2, ProjMan :CL2</td>
</tr>
<tr>
<td>PCP_b</td>
<td>SwReq :CL2, SwCons :CL1, SwTest :CL1, ProjMan :CL1</td>
</tr>
<tr>
<td>PCP_c</td>
<td>SwReq :CL2, SwCons :CL1, SwTest :CL1, ProjMan :CL1</td>
</tr>
<tr>
<td>PCP_d</td>
<td>SwReq :CL1, SwCons :CL1, SwTest :CL1, ProjMan :CL1</td>
</tr>
</tbody>
</table>

Note that these two hierarchies of PCPs have similarities and differences. Both of them are low capability PCPs use almost the same processes of software development. As the contexts were different, the sequences were different. These two hierarchies (Table 6 and 7) and the SW-CMM level 2 decomposition (Table 5) are not proposed for general case. They are appropriate for the specific contexts where they were defined.

### 5. Further work and conclusions

The best references for the work on 15504 are the three SPICE conferences [12,13,14]. Among the sixty-five publications, there are some examples of PCPs for assessment models and SPI projects, all of them defined in an informal way. Thus, there is no reported initiative for a methodology such as PRO2PI.

An approach for defining and applying target capability profiles for capability determination by integrators of component-based systems in providers of components is described in OOSPICE Project [11]. The approach is based upon the general model for Capability Determination described in TR 15504-8, makes use of defined Product-Process Dependencies, and expresses the results in terms of the process model defined by the OOSPICE Project.

PRO2PI is planned to be used in the 15504MPE research project, where a 15504 complaint process assessment model and method for process improvement adapted to small Brazilian software companies is being developed [1]. In addition to the required elements, the assessment method will include, in the planning phase, the definition of a PCP for each assessment, based on the process improvement goals.

We are confident that methodologies to support the usage of continuous models are necessary. Although more work needs to be done, there are evidences from the activities and results reported in this article that the PRO2PI methodology can play an important role to support an effective usage of third generation process capability frameworks, such as the new ISO/IEC 15504.

There are two planned actions to advance PRO2PI:

- a) improve the definition of its elements, including a representation in a formal notation, such as the Meta-Object Facility (MOF) [8], and
- b) continue the experimentation to guide SPI projects and analyze independent SPI projects.

A new version with these two evolutions is planned for July 2004.

### Acknowledgments

The authors would like to thank all the people from the SPI projects reported in this article and from CenPRA, and the anonymous reviewers for their comments and suggestions. Early work in this research has been
supported by a grant from CNPq, an entity of the Brazilian Government directed to scientific and technological development.

References


