I. Introduction

1. In recent years, the INE has made great efforts to transform the existing information systems based on "stovepipe" models to true statistical production models based on internationally recognised standards. The stovepipe model involves:

   (a) Independent systems where there are so many applications as surveys and collecting channels.
   (b) High costs due to outsourcing for development, and maintenance.
   (c) It takes long time to develop information systems
   (d) It's very difficult to reuse information or procedures.

2. We need to change from a stovepipe production model to a new statistical production system based on corporate databases and standardized tools.

3. NSI’s strategy to build the new production system is based on:

   (a) Building an integrated system.
   (b) Managing by standardized tools.
   (c) Using corporate Data Bases

   Allowing reuse tasks, processes and information
II. NSI’s new production system

A. Main elements

4. NSI’s new production system is based on the following elements:

(a) SADIS: System of Administrative data Integration for statistics.
(b) DS: Selective Debugging.
(c) DIA: Automatic detection and imputation system.
(d) CIP: Centralized information process
(e) IRIA: Centralized collect system

The difficulties involved in developing a system such as this mean that it is preferable to deal sequentially with each subsystem

B. SADIS

5. System to make use of administrative records, which aims to have a corporative micro-database linking databases to INE’s master-database (People, Addresses and Companies) with other administrative sources, including the INE itself.

6. SADIS will provide information fully integrated and available for all departments of the National Statistical Institute. We will use SADIS in the following phases of production surveys process:

(a) In stages previous to data collection. Having frameworks and samples of higher quality reducing burden on interviewees.

(b) In stages of information processing. Reducing the non-response rate or being able to treat it more carefully and to improve estimators and editing methods.

(c) In final stages In final stages SADIS will allow:

1. To design and develop surveys based on SADIS information with two strategies:
   a. Using partial information for adding it to an existing statistical source.
   b. Creating new surveys fully obtained from administrative records and / or other surveys.

2. To do analysis and studies with more and better information.

C. Selective Debugging

7. Selective debugging methods allow for a reduction in the times of re-contacts with the interviewed. For practical purposes, selective debugging methods (for certain confidence limits), determine which records should be edited manually.

8. Selective debugging is integrated in both manual editing (IRIA) and automatic editing (DIA).

D. DIA: Automatic Detection and Imputation System

9. DIA System (Automatic Detection and Imputation System) is a set of programs developed by the National Statistics Institute, used to debug statistical data using validation rules defined by users. The methodology that DIA follows is based on the theory developed by P. Fellegi and D. Holt and it meant a spectacular advance in relation to the way of detecting and correcting errors produced when statistical data are filled in.

10. The DIA version 2 processed only qualitative variables however the DIA version 3 (beta) processes all kind of variables (quantitative, qualitative and mixed).

1 In analysis phase
E. Centralized Information process

11. The stages of this system are most complex for standardization and will be studied in future plans. The design of this system, it is a basic requirement: The output of these processes will be stored in corporate databases.

III. Survey Generation and Information Collection (IRIA)

12. IRIA is a modular, configurable and extensible system that allows the end-user (without computer skills), to design, build, debug, exploit and manage all kind of surveys and reuse all modules or information with minimal involvement of the IT Department and generating a corporate database of raw data allowing natural integration of different types of information.

A. Main features

13. Modular and configurable: IRIA is formed by a pool of configurable modules. If some modules are selected and these are configured in a certain way, we will build a specific data collection application. These same modules differently configured will build a different data collection application.

14. Extensible: IRIA has several points where it is extensible without the need to develop or compile new code.
   
   (a) It is possible to automatically create web clients for use during an interview by means of a JavaScript function.
   (b) New components can be created for use in the design of questionnaires, and hence, in interviews.
   (c) It offers the possibility of extending the portlet catalogue used in the configuration of the screen displaying all the information on an informant.
   (d) It allows for the possibility of generating, incorporating and modifying the workflows that determine the sequence of tasks to be performed in each collection phase.

15. End-User: IRIA is a tool for the end-user (basic or advanced), therefore IRIA had been developed using the following guidelines:

   (a) Simple and user-friendly interface.
   (b) Has on-line help.
   (c) Allows us to do enough tests in order to be sure that the survey is free of errors.

16. All kind of surveys: With IRIA, the user will be able to design, build, debug, exploit and manage both social and economic surveys, both structural and short-term surveys and for all collecting channels (CAPI, CAWI, CATI and PAPI).

17. Reuse: Reuse is a key feature of the IRIA system. It covers the reuse of both components and information. Given that the IRIA system covers a range of surveys and questionnaires, reuse is essential for significant reductions in time to start gathering information.

   (a) Reuse of components: The MANAGER application has an interesting feature for copying diverse items from one survey to another, including the design of the questionnaire itself. Thus, it is quick and easy to obtain a survey with all of its items configured when it is similar to an
existing one. Moreover, libraries of different items are available for use, including question libraries.

(b) **Reuse of information:** Since IRIA is a unique system for all surveys, the reuse of information comes naturally to it. All surveys may use data from earlier periods of the same survey or data from other surveys or, indeed, data residing in other information repositories. This is accessed directly through a database query, web services, or by uploading additional information to the IRIA system.

18. **Minimal participation of IT Department:** According to our data, the participation of the IT department in the process of creating and implementing surveys is 35% and 65% in the phases of Design and Build of the survey. With IRIA these data change drastically. The IT participation in the whole procedure decreases to 6% and for the phases of design and build, only 8%.

<table>
<thead>
<tr>
<th></th>
<th>Specify Needs</th>
<th>Design</th>
<th>Build</th>
<th>Collect</th>
<th>Avg(Tot)</th>
<th>Avg(D,B)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Not IRIA</td>
<td>User</td>
<td>95</td>
<td>60</td>
<td>10</td>
<td>95</td>
<td>65</td>
</tr>
<tr>
<td></td>
<td>IT</td>
<td>5</td>
<td>40</td>
<td>90</td>
<td>5</td>
<td>35</td>
</tr>
<tr>
<td>IRIA</td>
<td>User</td>
<td>95</td>
<td>95</td>
<td>90</td>
<td>95</td>
<td>94</td>
</tr>
<tr>
<td></td>
<td>IT</td>
<td>5</td>
<td>5</td>
<td>10</td>
<td>5</td>
<td>6</td>
</tr>
</tbody>
</table>

19. **Integration:** This is another key feature of IRIA. Integration is not only possible with the other applications of the production system; it is also established as a key element:

20. **Integration of the data collected by the various collection mechanisms for each survey.**

   (a) Integration of all survey data into a corporate micro-database.

   (b) This corporate collection micro-database is a key element for integration with other subsystems.

Through this micro-database, the global data from all surveys is made available to the users who require it, whether independently or through the record exploitation system. Lastly, the micro-database is the starting point for the performance of all general exploitation tasks, closing the cycle of statistical production.

IV. **General sub-systems**

21. IRIA currently consists of four general sub-systems:

   (c) **iriAMANAGER:** for the management of surveys.

   (d) **iriDESIGNER:** for the design and testing of questionnaires.

   (e) **iriENGINE:** executes the interviews designed in DESIGNER.

   (f) **iriDATACOLLECTION:** To control and manage the execution of the survey by the different collection channels (CAPI, CAWI, CATI and PAPI).

   **A. iriAMANAGER**

22. This is the system management sub-system. Its main features are:
(a) **Three user levels:**

a. **System administrator**: manages basic elements (users, roles, table maintenance, etc.).
b. **Survey manager**: creates and maintains the items in each survey: survey properties, periods, survey users, collection units, tasks to complete in the survey, aspects of collection, inspection, coding, etc.
c. **Unit manager**: manages and fills IRIA tables with units that are then assigned to the various surveys (households, individuals, economic units and other entities).

(b) **Modules:**

a. **Security**: The system administrators create user roles and assign specific permissions to each, allowing certain permissions to be extended or restricted to individual users. With this system, user management is performed hierarchically, thus redirecting these tasks to the users who manage the surveys.
b. **Survey manager**: Module for managing surveys and assigning their properties and the properties of each collection mechanism. Survey management covers the management of the collection periods, the properties of each period, tasks to complete in each survey, etc. Survey management also includes the management of permitted INE user accesses and the distribution of units among the different collection units involved in the process.
c. **Load information**: For uploading information to the system, offering a straightforward way of adding sampling units, assigning units to surveys, determining which units are assigned to each collection period or simply uploading additional information from administrative or other sources.
d. **Workflow management**: It has a module for incorporating workflows in order to determine the flow of work required in the survey (statuses that each interview must go through, tasks to complete, etc.). Each workflow transition between two statuses includes the logic on the preconditions that need to be met in order to make the transition and actions to take during or after the transition. Each survey can have one or more workflows that call one another. All workflows merge into a main workflow.
e. **Design**: To define the screen in which each sampling unit is to be managed, installed as a sum of portlets, which means that each survey can have its own design. The portlet catalogue can easily be expanded because it is written in a non-compiled language.

(c) **SQL, Web Service and functions libraries**: In general IRIA allows reusing all kind of components or information of system. In addition iriaMANAGER has a module for libraries and specialized services management. These libraries are:

a. **SQL queries library**: Catalog of Database queries to generate lists associated to questions, generations on OLAP cubes, etc. They can be reused to generate lists associated to questions in the iriaDESIGNER application.
b. **Functions library**: Catalog of functions to give logic to expressions, validations, workflows etc. to get the desired behavior of the interview.
c. **Web Services library**: The web services library allows calls to services to be made during interview execution in order to retrieve the information needed to generate a question, perform a validation. The web-services can be predefined and automatically create client Web Services.

**B. iriaDESIGNER**

23. This allows questionnaires to be designed, built and tested with a user-friendly interface. Its main features are:
(a) **Three user levels:**

a. **Basic**: creates questionnaire structures and questions using the basic interface features. This includes creating validations and simple filters.
b. **Advanced**: assigns advanced properties to questions and other elements and can be used to introduce SQL statements, functions and expressions to create more complex structures.
c. **Expert**: accesses the questionnaire creation code (XML) and can import externally generated questionnaires to the tool.

24. For practical purposes iriaDESIGNER distinguishes: Structure of the questionnaire and global items, which are those that can be applied to all or part of the questionnaire.

25. A questionnaire is made by (Structure): **Questions**: minimum level of the questionnaire. **Screens**: The questions are grouped into screens. **Blocks**: The displays are grouped into blocks. A block can be 1 or "n" screens. **Forms**: The forms are grouped into blocks. A form can be formed by one or "n" blocks. A questionnaire can be formed by one or "n" forms. Information **flows** between questions, screens, Blocks and Forms and **Validations**

26. Global elements: Each of these elements of a questionnaire must have specific properties, among the main properties are: **Languages** used in the interview. **Functions**, which can be obtained from the library or created from scratch. **Templates** to improve or adjust the appearance of screens and interview questions. **Style** guides. Use of general **messages**. Other items, such as database connection, the definition of the toolbar, link to Help, etc.

27. Questions are divided into four basic types: **simple, multiple, quantity** (string, numeric, date...), and **grid or table** type. The questions can be obtained from the corporate metadata system.

28. One of the most interesting features of iriaDESIGNER is the possibility of displaying these items in the navigation menu, allowing the user to return directly to previous screens. The screens also have a Preview feature, so the user designing the questionnaire can view the on-screen layout, and application of the styles applied.

29. Questionnaires can be run in demo mode as they are designed. This mode includes an editing module for users to check the values taken by the different variables, change values, verify the correct behaviour of features, flows, validation, etc.

30. The user will be able to design, build and test questionnaires:

   (a) Reusing any existing component from IRIA (questions, screens, flows, styles, templates, libraries, etc).
   (b) The questions can be obtained from the corporate metadata system.
   (c) Creating any not previously existing component.

31. The required steps for designing a questionnaire are:

   (a) Add the required global items.
   (b) Add the questions, which are obtained either from the corporate metadata system or as new questions.
   (c) Add the logic needed to obtain the desired interview behaviour: flows, validations, etc.
   (d) Add appearance-related elements: templates, style sheets, etc.
   (e) Test the questionnaire. As the questionnaire is designed, you can run it in test mode to verify the value of each variable, change values, verify behavior of functions, information flows, validations etc.
Since this tool can be used to design any questionnaire, it will be used to generate other "questionnaires" to be used in iriaDATACOLLECTION, such as repeat interviews, coding, and the management questionnaire, which is the questionnaire that leads the interviewer to an interview state.

C. iriaENGINE

This is responsible for executing the interview and interpreting the elements and properties assigned in iriaDESIGNER. It controls the interview from start to finish, from when the user enters and navigates, to its completion or cancellation.

iriaENGINE stores the information entered by the user but also saves other information of interest, often for subsequent studies of quality:

(a) The time spent on each screen.
(b) User navigation.
(c) The data initially and ultimately entered by the user.
(d) Validation errors detected for the first time and the final errors included in the questionnaire.

Since the IRIA system has been designed for use by staff without knowledge of computers, it is not free from errors in functions, database queries, etc. not detected in the design phase. IriaENGINE is therefore a highly protected module with comprehensive error control to keep the interview running where possible. It also sends e-mail alerts of the errors detected during execution of the interviews to administrators.

D. iriaDATACOLLECTION

The iriaDATACOLLECTION subsystem can be used to perform and control all of the activities conducted during the data collection phase.

Knowledge and control of situation of collecting survey. The user can know in real-time the situation of the survey. The control information is generated from pre-existing queries or created by the user himself. All of them will be stored in libraries to be reused by other users. To get this kind of information, the user can use business Intelligence utilities such as OLAP cubes (iriaMONDRIAN) or SAS_VA.

Knowledge of status of each questionnaire. iriaDATACOLLECTION uses the workflows designed in iriaMANAGER, allowing the user to use pre-defined queries or new queries to access the real situation of each questionnaire. Besides questionnaire status, the user can get other features such as incidents during interview attempts, the interview itself, its coding, inspection status etc. Finally the user can know the roadmap with all the history of actions performed on a collection unit.

Distribution and control tasks: Depending on the permissions of user, he may have control over one or more collection units, one or more surveys and for one or more collecting channels for:

(a) Assigning or reassigning collecting units to a particular interviewer both by direct assignment or queues/quotas.
(b) Allocation of managements to users or assigning user hierarchies over another.
(c) Assigning management to more than one person for different tasks.
(d) Replacing interviewers on vacations, medical leave, etc.
(e) Getting reports of tasks assignment
(f) Consulting, updating and prioritizing tasks interviewer Agenda
40. **Other utilities:**

   (a) **Permissions**: Permissions are assigned hierarchically. Each role has its own permissions and can grant permissions. Each user has the permissions of their role, but may have specific permissions.

   (b) **Messaging**: iriaDATACOLLECTION has a messaging system integrated with the security module that can be used to send messages to survey users, collection units, etc.

   (c) The user can access the **digital information** generated by the collection channel PAPI.

   (d) **Communications with informants**: For designing and editing mailing templates, label generation, letters and emails.

V. **Specialized Sub-Systems**

41. So far we have seen the general subsystems that allow designing, building, configuring surveys and questionnaires and even performing an effective control about the survey and each questionnaire. But none of this is possible without the presence of subsystems specialized in the collection of information itself. These subsystems are the following:

   (a) **CAWI channel**: IriaPORTAL

   (b) **CAPI channel**: IriaCAPI, iriaCAPISYNCHRONIZATION and iriaWS (and iriaJNLP), for devices control, loading and unloading data, synchronization, etc.

   (c) **CATI channel**: iriaCATI, iriaCTI (and iriaJNLP) for telephone integration with algorithm calls.

   (d) **PAPI channel**: iriaSCAN, for self-administered paper, managing shipments, receipt paper, scans, etc.

A. **IriaPORTAL**

42. IriaPORTAL allows filling the interview by CAWI channel and allows the user to know the status of processing of their surveys, has been designed to provide general information on the situation of sampling units in each survey, regardless of the collection mechanism used to conduct the interview. IriaPORTAL is very straightforward but has some interesting features:

   (a) Access is afforded by a key pair that is unique to each interview, although users may also be registered for access to display information from one or more selected sampling units, or certificates may be used to identify users.

   (b) For economic surveys, where a sampling unit is selected in several surveys, registered users or those with a digital certificate can obtain information on the units, the surveys associated to each, and information on completed or pending interviews. Registered users can also receive e-mail alerts when the term for completing an interview begins.

B. **IriaCAPI**

43. Adaptation to the general questionnaire collected by CAPI channel and other utilities. iriaDATACOLLECTION, iriaENGINE, IriaCAPI, iriaWS and iriaCAPISYNCHRONIZATION are joint executed to get the functionality that the user needs to:

   (a) Perform CAPI interview.

   (b) Synchronize your data with the central (remote) database.

   (c) Take GPS positions.

   (d) Know the route to follow.

   (e) Know the location of the collection unit.

   (f) Make notes and appointments on your calendar.
Send and receive user alerts
Know the status of the battery.
Etc.

C. iriaWS

Set of web services for interaction with portable devices with central database as the user login in the corporate personnel system or registration of the portable device in the synchronization system.

D. iriaCAPISYNCHRONIZATION

This module is responsible for synchronizing information between tabletsPC and central database, for that it uses an open source tool, SymmetricDS.

The parameterization of this tool determines that information should be delivered from the server to client and from client to server (bidirectional) with both client and server recording the status of each transaction which allows knowing the status of a local database.

It also incorporates conflict control, for cases where the expected timing information does not match what exists at the moment.

E. iriaCATI

Adaptation to the general questionnaire collected by CATI channel and other utilities. For the execution of a CATI interview, besides the form, control and implementation of the questionnaire itself (IriaCAPI, iriaDATACOLLECTION, iriaENGINE), it is required to have the necessary elements for the integration of the interview in a telephone system CTI (Computer Telephony Integration) and call recording.

It has an algorithm of calls that determines the next call to be made. IriaCAPI communicates with the CTI server to start the call. IriaCAPI has a toolbar allowing the user to perform the usual actions of an operator such as hang, hold, transfer a call, etc.

IriaCAPI listens for events from the CTI server to give each user the information that corresponds such as incoming call, busy, no answer, etc.

It also has the ability to access in monitoring mode, which allows knowing the status of telephone operators, listening to active calls, record calls, access recorded calls, or display the operator.

The connection between IriaCAPI and the CTI server is made by IriaCTI.

F. iriaCTI

The connection between IriaCAPI and the CTI server is made through IriaCTI. IriaCTI is invoked by IriaCAPI using Web Services so that when iriaCATI needs to order something from CTI server, it sends a JMS message to iriaCTI which interprets the message and executes the corresponding Web Service of CTI server.

The answer of CTI server is sent to iriaCTI which sends a JMS message to iriaCATI for ending the communication.
G. **iriaJNLP**

55. iriaCAPI and iriaCTI applications are installed on PCs. When a change occurs in these applications, these are stored in a centralized repository of applications. iriaJNLP is the application responsible for downloading the software required from server to PC.

56. Besides, iriaJNLP manages the software that corresponds to each user. The software is grouped into kits and each user group is associated with a kit, calculating the differences between what is already installed locally and new software that might correspond to each user.

57. Finally JNLP allows downloading of an additional application, iriaCAPIADMIN, to recreate the situation of a user of a portable device.

H. **iriaMONDRIAN**

58. The user can perform a check on the status of the survey through listings and predefined queries or conducted ad-hoc or by the use of OLAP cubes generated automatically in IRIA. The generation and access to these cubes is performed by iriaMONDRIAN which uses Mondrian, a library of Pentaho, customized for the INE.

I. **iriaSCAN**

59. Currently iriaSCAN is a subsystem that enables scanning and digitization of questionnaires (using OCR techniques with or without operator intervention) and their integration with the rest of the collecting channels.

VI. **Manual Debugging**

60. An important feature of IRIA is the ability to run the questionnaire in debug mode. The user can establish a set of validations on survey collection and another different set of validations to consider that a questionnaire is valid and may be processing information so a questionnaire can be considered valid in the collection stage and not valid for processing, implying a debug thereof.

61. This debugging in the INE’s current production system of statistics can be performed in conjunction with selective debugging methods, automatic detection and imputation methods and finally manual methods with the use of the debugging mode of IRIA.

VII. **Technological environment**

62. The IRIA tool is a full web application developed in Java 6 that uses the following technologies in the applications already in place:

(a) Presentation JSF 1.2, Facelets, Richfaces, JSP 2.1, Servlet 2.5
(b) Persistence JPA 1.0 (OpenJPA 1.2.2), JDBC
(c) XML XML, XSLT, XSD, JAXB
(d) Web services AXIS, JAXB-WS

63. Its implementation for the INE used JBoss EAP 5.1 application servers and an Oracle 11gR2 database, although the database does not contain any elements specific to the platform, so any relational database could be used. In the second development phase (implementation of mechanisms other than CAWI), the development library versions have been updated: JSF2.0, JPA2.0, JBoss 6.1 EAP.
With regard to the potential users of the tool, from basic to advanced level, they can use:

(a) CSS and HTML. The tool allows the use of HTML tags and the addition of styles to overwrite the ones configured originally.

(b) Facelets, allowing questions to be positioned on-screen (by default, these are always placed in a certain position) and allowing the user to position on the screen specific messages indicated in Design or actions such as the evaluation of a JavaScript function included in DESIGNER.

(c) JavaScript (server side), for giving implementation logic to the questionnaire, the evaluation of expressions, functions, and so on, used for the validation of data, flow control, etc.

(d) The ENGINE application offers an API for use in JavaScript, for the dynamic evaluation and control of interviews. It has properties and methods for each question type, for each of the structures making up a questionnaire, and properties and methods for the interview controller.

Just as it is possible to use server-side JavaScript (ECMA specification) to evaluate expressions, we can use the Java libraries from JavaScript functions. The methods and properties of the objects in the Java libraries can therefore be used, although it is also possible to add new libraries, developed for specific uses, which can be used from JavaScript functions.

VIII. Adaptation of IRIA to GSBPM

The adjoining figure shows the GSBPM model, indicating the sub-processes fully supported by IRIA in green, partially supported sub-processes in orange and those for which IRIA may be useful in yellow.

In addition to adaptation to the phases and sub-processes indicated below, iriaMANAGER module has specific tools to meet all GSBPM Level 0 administrative requirements.

(a) PHASE 1. SUB-PROCESSES 1.1 and 1.2.

With regard to the possibility of producing complete prototypes for new surveys or making improvements to the same, IRIA is a good support tool in the "Specify needs" phase.

(b) PHASE 2. SUB-PROCESSES 2.2., 2.3. AND 2.4.

IRIA can provide information on national or international standards and can be integrated with other information systems or retrieve data or classifications via web services. IRIA allows the creation of repositories of questions or groups of the latter that are common to several surveys (e.g. household composition and questions related to each household member). This ensures quick and easy access to tools for cognitive testing or for the performance of pilot collection tests. IRIA also maintains a history of the units that have been selected at some point in a survey, allowing for the reuse of this information; this, combined with the possibility of using information from other sources, should allow the optimal design of questions and even support for decisions on sample selection.

(c) PHASE 3. SUB-PROCESSES 3.1., 3.2., 3.4., 3.5. AND 3.6.

IRIA offers a straightforward and user-friendly method of constructing the collection tool. It also allows new components to be created in order to design questions to suit the collection mechanism or features of different surveys. Surveys are available for the performance of pilot tests to check both the tool and the structure or formulation of the questions designed. As a unique tool, all necessary user manuals will be made available, which will be reused in the different surveys. User training will be fast and stable over time, since the same work tool is always used.
(d) PHASE 4. SUB-PROCESSES 4.2., 4.3. AND 4.4.

Obviously, in this phase, the framing of the IRIA system is more advanced, which is why its adaptation to GSBPM is considered to be virtually complete. With regard to sub-process 4.3, much of the analysis of the development to be undertaken over the coming months has already been completed, including the management of interviewers with reporting units and, hence, the monitoring, control and exploitation of the field work.

(E) PHASE 5. SUB-PROCESS 5.3

In all cases, the surveys developed with IRIA include the pre-validation of responses and give the respondent the opportunity to edit these, which means that the data collected by the applications developed with this system have been validated previously.