

Spreadsheets: From Data Interfaces to Knowledge Interfaces

Andrea Kohlhase

Jacobs University Bremen

Abstract. Documents of type “spreadsheet” are considered user interfaces to numeric data as they allow authors to create, modify and display these data in distinct layouts like tables or diagrams and readers to interpret them. We tend to believe that enhancing software semantically means that we are lifting its value. In particular, if we enhance spreadsheets semantically can we lift their data interface status to a knowledge interface status? We used the repertory grid methodology to conduct a study on the difference between spreadsheets and spreadsheets semantically enhanced with the *SACHS* extension. Our research shows that, indeed, from the perspective of users adding semantics turns spreadsheets into knowledge interfaces.

1 Introduction

The document type “spreadsheet” is indeed very successful: tens of millions professionals and managers create hundreds of millions of spreadsheets according to [15]. Spreadsheet documents are used to create, modify, and visualize numeric business and science data, therefore they are mathematical user interfaces. Their complexity and impact increased at the same time: this intensity yields wide-impact errors on the data level (up to 90% [15], see also [17]) and on the apprehension level (e.g. [16]). To address this problem, in the past we have extended spreadsheets semantically with the “*SACHS* (sx)” system [12, 13].

In this paper, we ask what difference *SACHS* really makes: Do people perceive a difference when offered *SACHS* functionality in spreadsheets¹?

To better understand what users of spreadsheets perceive commonly as information units, what meaning users assign to these units and what meaning users assign to the additional *SACHS* information units, and finally, how users discriminate between them, we conducted a study using the **Repertory Grid Interview (RGI) Technique** [11, 7, 10]. RGI explores personal constructs, i.e., how persons perceive and understand the world around them. It has been used as a usability/user experience method to research users’ personal constructs when interacting with software artifacts (see [9, 19, 8, 4, 21] for examples). RGI is a semi-empirical method that has the grand advantage of not depending on a high amount of study subjects and nevertheless delivering valuable insights into the perception of users.

¹ At this point we still use the terms “spreadsheets” and “spreadsheet application” unspecifically, because the concepts involved will only be clarified by the research reported on in this paper.

As a pre-study we used a smaller RGI to elicit which “semantic objects”, i.e., meaningful objects, are commonly perceived by users in spreadsheets. From these we identified those semantic objects, which were considered information objects. For our main study we added information units provided by SACHS. Besides getting a better grasp on human-spreadsheet interaction in general, we were specifically interested to find out which of these external information objects were perceived to deliver similar or different information compared to traditional spreadsheet information objects.

First we present the setup of our repertory grid interviews. Then we analyze the RGI and interpret the results. Finally, we conclude by drawing conclusions from our RGI study.

2 The RGI Study

The aim of the study is a better understanding of information quality within a spreadsheet and whether existing applications already cover the information offerings of the SACHS extension. If such additional information were to be perceived by readers at all, we were especially interested in what ways these new qualities would be perceived.

A **repertory grid** is a grid consisting of “**elements**”, i.e., the objects under consideration, and “**constructs**”, i.e., pairs of antithetical properties that separate elements. The constructs serve as a bipolar dimension on which the elements are evaluated. As the property elicited first in a construct is the more salient one, RGI calls it the “**implicit pole**” and the other one emerging in the reflection of the dimension of comparison the “**emergent pole**”.

Elements as well as constructs can be elicited by the test persons themselves or can be provided by the interviewer. Comparison of multiple repertory grids is simplified if the individual ratings are given on a fixed set of elements or constructs, but a free elicitation explores the perception space. For our main RGI we decided to fix the set of elements to be “common and additional information objects in spreadsheets”, but to elicit individual constructs to better understand the information space.

In a first RGI we extracted spreadsheet elements that are considered common information objects for the main RGI. Then we selected six additional spreadsheet information objects from SACHS offerings, that are not contained in a common setup with spreadsheets. We used both element sets together to collect personal constructs used to evaluate these elements. Concretely, we asked the interviewees to tell us in what ways the selected elements considered as information objects were similar and different (according to traditional RGI).

2.1 Common Information Objects in Spreadsheets

We asked users to mark those elements from their self-created list of elements in the first RGI study, which they consider to be “information objects”. We explained that we understand **information objects** to be objects carrying information identifiable to the reader.

Even though the resulting principal components given by a factor analysis and an accordingly weighed clustering were interesting per se, they wouldn’t give us a gener-

ally valid assessment, as the constructs were not directly comparable. But we found six information elements in this small RGI to be consistently listed:

Title	A phrase describing the content of the spreadsheet
Headers	A (short) phrase supporting the interpretation of values of a regionally close range of cells (e.g. a column header)
Legends	A list of content properties and resp. layouts (as in a map legend)
Values	The content of a cell container
Formulae	A computational rule that yields a cell value
(sx:)Color Coding	The use of color hinting at additional information

Furthermore, two subjects also identified:

Tables	A possibly multidimensional homogenous structural layout of cells, that is perceived as an object of its own
--------	--

Note that “diagrams” are missing, which may be due to the fact, that they were not part of our standard spreadsheet example.

We selected these 7 information objects of common spreadsheets as part of the set of elements for the main RGI.

2.2 SACHS Information Objects in Spreadsheets

The SACHS system contains semantic spreadsheet-related information not usually available to spreadsheet users. It aims at providing user assistance for spreadsheet users based on a background ontology. As “cells” are the important semantic objects in spreadsheets, SACHS acts cell-oriented.

In a spreadsheet like the one in Fig. 1 a user clicks for example on a cell that contains “1,878” as information of Values. Common information objects tell the user about the context (e.g. by Title “Profit and Loss Statement”, Headers “Profits” and “1986”, or Legends “in Millions”). But what if the user doesn’t understand how “Profits” are calculated? When using SACHS this cell might be linked to a concept in a background ontology that covers the domain knowledge of this particular spreadsheet. If the user likes to retrieve this linked concept from the ontology, then she can do so by opting for a “look-up” option provided by SACHS and by selecting the wanted cell. A pop-up close to the selected cell will appear with this additional information — e.g. “A profit is the difference between revenues and expenses.” — together with the header information “Profits [1986]”.

	A	B	C	D	E	F
1	Profit and Loss Statement					
2						
3	(in Millions)	Actual		Projected		
4		1984	1985	1986	1987	1988
5						
6	Revenues	3,865	4,992	5,803	6,022	6,481
7						
8	Expenses					
9	Salaries	0,285	0,337	0,506	0,617	0,705
10	Utilities	0,178	0,303	0,384	0,419	0,551
11	Materials	1,004	1,782	2,046	2,273	2,119
12	Administration	0,281	0,288	0,315	0,368	0,415
13	Other	0,455	0,541	0,674	0,772	0,783
14						
15	Total Expenses	2,203	3,251	3,925	4,449	4,573
16						
17	Profit (Loss)	1,662	1,741	1,878	1,573	1,908
18						

Fig. 1. A Simple Spreadsheet after [20]

For this study, we extracted the following information objects that seemed to carry *additional* information benefits compared to the common information objects:

<code>sx:-</code> Localized Info	A local look-up (data and text) of relevant information for cells on a by-cell-click basis
<code>sx:Function-</code> al Block	A local border indicating all cells functionally associated to the currently selected cell
<code>sx:Dependen-</code> cy Graph	An overview graph (in a different window) of concepts showing on which the corresponding (selected) cell is ontologically dependent
<code>sx:Relatio-</code> nal Arrows	An arrow indicating a dependency relation between concepts in <code>sx:Dependency Graph</code>
<code>sx:Concept</code> Nodes	A node in <code>sx:Dependency Graph</code> representing a dependent subconcept, that additionally serves as a link to corresponding spreadsheet cells

2.3 Data

For our study we interviewed 14 people, of which 10 were male and 4 female. The age mean was 29,3 years. Participants reported an average of 8.2 construct pairs (SD = 1.4) ranging between 5 and 11 pairs. 5 subjects were familiar with authoring spreadsheets, the other 9 only had occasional contact. The rating scale for the 115 elicited constructs was essentially binary: it consisted of -1,0,1 but the interviewees were only told about their option to use “0” as a rating when they otherwise would have discarded the construct in question as inapplicable.

We performed a Generalized Procrustes Analysis, as it can be used when data “*have arisen from one type of scaling of the same stimuli as perceived by different individuals*” [5, p. 33]. In particular, in our RGI we can compare the individual natural language constructs rated on our fixed set of information objects. We follow Grice’s example procedure for a Generalized Procrustes Example [6]. In particular, after having produced a “**consensus grid**”, i.e., a best fit grid for a number of grids that are equal in one dimension but not in the other, we conducted a **Principal Components Analysis (PCA)** on it yielding components $\{PC_{i=1,\dots,11}\}$. The first two components explain ca. 56% of the variance in the data, the first three even 71%. Hereafter, a **Multiple Group Components Analysis** was performed, which allows to map the specific construct/element distribution into the PCA results.².

3 Analysis and Interpretation

We now give an overview of element clusters we identified and the outcome of the reconstruction of elements/constructs via the Multiple Group Components Analysis of all elicited repertory grids followed by an interpretative discussion of the findings.

² The original subject and computed data are available at <http://kwarc.info/ako/ProcrustesAnalysis>

We focused each repertory grid by swapping the construct poles to optimize the amount of applicable poles for the set of common spreadsheet information objects. This way we could identify the characteristic construct poles for this and the remnant element set and their pole distribution.

For the analysis of *multiple* repertory grids we found the “Idiogrid”³ tools of analysis most helpful. Especially the availability of the Generalized Procrustes Analysis [5] resulting in a consensus grid, yielded a valuable input for other analyses. The constructs of the consensus grid are created based on the underlying statistical analysis, thus, they are an artificial gold standard for real elicited constructs.

In Fig. 3 we can see the outcome of the Multiple Group Components Analysis for PC_1 and PC_2 (with only the more salient constructs, in particular with 0.85% suppression of labels) run by Idiogrid. Emergent Poles are marked by a “(-)” prefix.

3.1 Elements

Let us first look at the elements themselves. As the set of elements were fixed in this RGI, we could build the

“concatenated grid” consisting of the given elements and their ratings on *all* elicited constructs. For this concatenated grid we ran a cluster analysis in OpenRepGrid⁴. Its dendrogram visualization (to be seen in Fig. 2) identifies three clusters. Note that these clusters can also be easily spotted in Fig. 3. They can be described as follows:

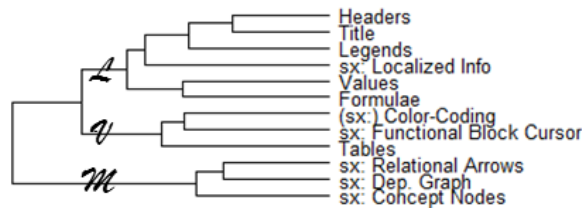


Fig. 2. Element Cluster Dendrogram for the Concatenated Grid

Local Cluster (\mathcal{L}) The first cluster mainly contains the elements in quadrant II and III of Fig. 3. Concretely it consists of the elements Headers, Title, Legends, `sx: Localized Info`, Values, and Formulae. This cluster contains all local spreadsheet information objects — those elements whose content depend on their position. Therefore, we can categorize this cluster as the locally perceived information objects group or in short the “*local cluster*”. Headers, Title, and Legends build a nested cluster as well as Values together with Formulae. Both are linked in the main cluster via `sx: Localized Info`.

Visual Cluster (\mathcal{V}) The second group contains all elements in the first quadrant of Fig. 3, specifically the elements `(sx:) Color Coding`, `sx: Functional Block`, and Tables. As all of them communicate information visually, we call it the “*visual cluster*”.

Meta Cluster (\mathcal{M}) In quadrant IV of Fig. 3 we find the remaining objects: `sx: Relational Arrows`, `sx: Dependency Graph`, and `sx: Concept Nodes`. The

³ <http://www.idiogrid.com/>

⁴ <http://www.openrepgrid.uni-bremen.de/wiki>

most salient pole nearest this cluster that does not refer to their property of being external to MS Excel reads “meta level”, therefore we call this group the “*meta cluster*”.

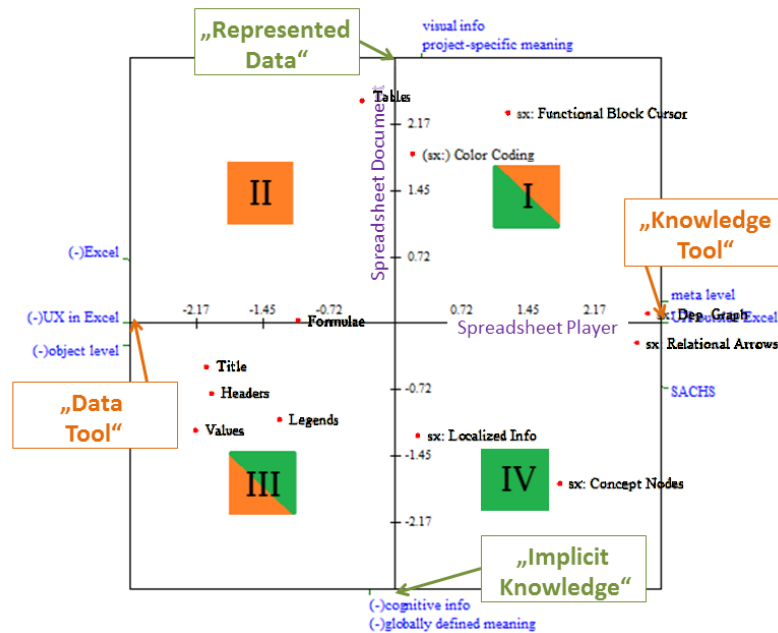


Fig. 3. Multiple Group Components Analysis for PC_1 and PC_2

3.2 Constructs

To approximate the meaning of the principal components, we looked at actual, similar constructs, especially at the more salient ones close to the axes in Fig. 3. Then we tried to find categories that can serve as common denominator constructs. As this content analysis was qualitative, the reliability was ensured by following the procedure given in [10, 155ff.].

Principal Component PC_1 The constructs coming closest to the first principal component (depicted by the horizontal axis in Fig. 3) are:

Implicit Pole “Knowledge Tool”	Emergent Pole “Data Tool”
meta level	object level
UX outside Excel	UX in Excel
implicit info	explicit info
relevant for analysis	relevant for understanding
represents relational info	represents contextual info

Here, the black entries are more salient than the gray ones (cited for clarification). The main property which the poles of this component share is that they classify the elements according to how they are used by a spreadsheet user, i.e., according to their purpose.

Probst et al. suggested in [18] a knowledge management model positing that glyphs, data, information, and knowledge can be seen as stages of a pipeline as in Fig. 4. This

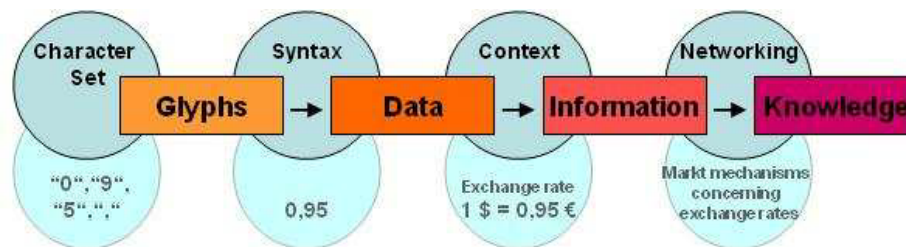


Fig. 4. Knowledge Management Model after [18]

model differentiates what we have simply called “information” so far into four distinct traits. Glyphs are just a set of characters without any structure, combined with a syntax they become data, additionally enriched by context they become information, and finally, they turn into knowledge if a semantic net or a global context is present.

Even though Brown & Duguid don’t speak of knowledge as such in their model, they also suggest to go beyond information in [1]. What this knowledge might be for knowledge workers is discussed in [2]. In the following we use the four distinct traits of “information” established by Probst et al.. We will use the term “information” in its generic (naive) form as before.

For the descriptions of the implicit poles this means that the information objects were used as “**Knowledge Tools**”, i.e., that the communicated information aims at the knowledge level. In contrast, the descriptions of the emergent poles indicate that the according information objects are being used as “**Data Tools**”: they handle information as pure data. We can say that in this categorial construct a *macro perspective* on information objects is captured.

Principal Component PC_2 The second component (vertical in Fig. 3) can be described best by the following constructs:

Implicit Pole “Represented Data”	Emergent Pole “Implicit Knowledge”
visual information	cognitive information
project-specific meaning	globally defined meaning

When we looked for properties which the implicit resp. emergent poles of the second component shared, we were surprised to find the elements of Probst et al.’s knowledge management model again. Here, a *micro perspective* is taken up in the interviewees’ constructs. They target what the information object itself uses and makes use of, that is, the categorial construct is concerned with what the content of the elements pertains and how it does so. The implicit poles we categorized as **“Represented Data”** whereas the emergent poles could be summarized as **“Implicit Knowledge”**.

Principal Component PC_3 To also touch the third principal component we analyze the constructs closest to PC_3 in the biplot of PC_1 and PC_3 resp. PC_2 and PC_3 , which are at suppression of labels at 0.78:

Implicit Pole “Creator”	Emergent Pole “User”
computation	explanation
data processing	semantics

Both, “computation” and “data processing” are concerned with the spreadsheet as intelligent application, but also with the author who uses the application to handle her input data. Thus, the implicit pole can be summarized under **“Creator”** as either application or author produces the spreadsheet by computation and by data processing. On the other hand, “explanation” and “semantics” apparently refer to the meaning of the document. To contrast it with the implicit pole we therefore named the emergent pole **“User”**, which is justified since meaning of software artifacts can only be evaluated and experienced by its users. Please note that creators are applications as well as authors, whereas users can be authors as well as readers.

3.3 Discussion

Let us now combine the findings about the elements and the constructs. Note that the term “versus” in the subtitles does not signify opposition, but is supposed to enhance users’ distinct context experiences implied by our subjects’ construct elicitation.

Common Spreadsheets versus SACHS Extension If we look at the element space with the coordinate system slightly shifted according to the construct vectors nearest to the axes, then *all* common spreadsheet information objects are on the left side and *all* SACHS ones are on the right side (except for the hybrid (sx:) Color Coding which is located very close to the separating axis). The construct “object level - meta level” is one of the constructs for which rating of the elements differ: common spreadsheet objects are considered to be at the object level, whereas SACHS objects are considered to be at the meta level. Similar constructs directly distinguish standard objects from

the additional ones (e.g. “UX in Excel - UX outside Excel” or even “Excel - SACHS”), thus the term “meta” could indicate a mere “going beyond”. Therefore we looked at more distinguishing constructs nearby by looking at Fig. 3 from within Idiogrid with suppression 0.70 and found e.g. the following:

Implicit Pole “Common Spreadsheet” (→ “Data Tools”, PC_1)	Emergent Pole “SACHS” (→ “Knowledge Tools”, PC_1)
structures info	structures meaning
represents contextual info	represents relational info
explicit info	implicit info
relevant for understanding	relevant for analysis
not formal info	dependency info
global context info	interpretative info
visual interpretation help	structural interpretation help

Thus, we can conclude that our subjects perceived a qualitative difference between common spreadsheet- and SACHS information objects, where former are concerned with supporting the data interface qualities, i.e., on the object level, whereas latter aim at providing a global context as interpretation help for the data interface, that is on the meta level.

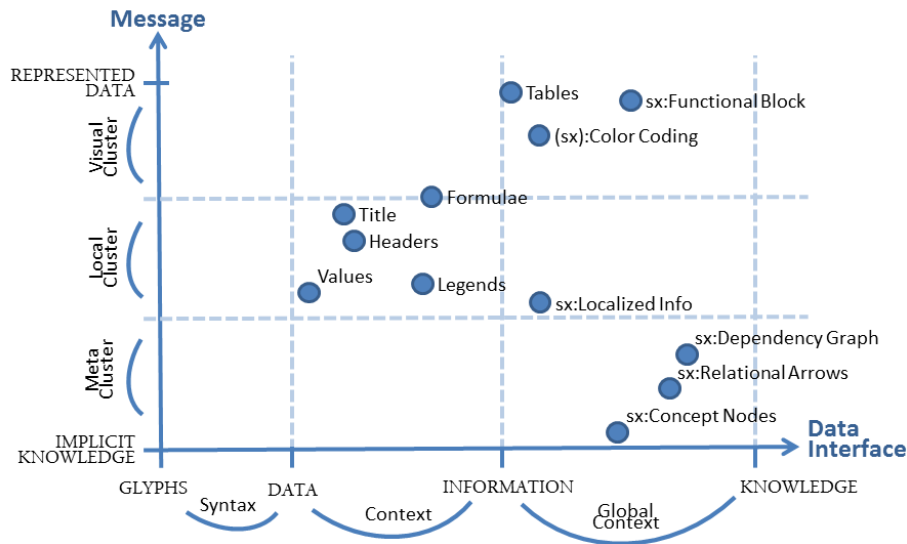


Fig. 5. Interpretation of Fig. 3 and Fig. 2

Spreadsheet Player versus Spreadsheet Document At first glance it was troubling that the properties of the first and the second principal component both concerned “data” and “knowledge”, as they were supposed to stress differences between elements. But a closer look (presented subsequently) reveals that the macro perspective “Knowledge Tool — Data Tool” rates the elements as information objects of the player whereas a view from the micro perspective “Represented Data — Implicit Knowledge” considers the elements as information objects of the document. A **spreadsheet player** is the application (e.g. MS Excel) which acts as a smart interface between the spreadsheet workbook and the human reader. This player e.g. renders the workbook on screen, interprets cells as value containers, runs computations for “cells”, and so on. In a sense it is an interpreter/compiler for input data. This input is given by a **spreadsheet document**. The document is an intermediate data layer as it presents a prepared view on the real data e.g. stemming from a database. Note that players “play” such a pre-compilation of data, whereas documents “document” (= “to furnish documentary evidence of, to provide with factual or substantial support for statements made or a hypothesis proposed” [14]) data.

The information objects perceived as “Data Tools” are Title, Headers, Formulae, Legends, and Values. They either describe data coming from some input source or work on such. They serve the purpose of enriching these input data with context, thus elevating them into information. The spreadsheet player supports the transition from data to information e.g. by simply distinguishing between various cell formats (like “text” for Headers or “date” for Values) and by computational tools for Formulae. As “Data Tools” these information objects thus belong to the *document player*. With the same reasoning the elements perceived as “Knowledge Tools” are information objects viewed as spreadsheet player objects (turning data into knowledge). For example, if we look at the visual cluster \mathcal{V} from the player perspective, they serve as “Knowledge Tools” as they help the user to elevate information into knowledge.

In Fig. 5 we visualized the element distribution according to the Principal Component Analysis as in Fig. 3. The only difference is that we enhanced the distance between the element clusters \mathcal{L} , \mathcal{V} , and \mathcal{M} determined in Fig. 2 to allow for a horizontal grid to depict the distinctions discussed in the following. In light of the discussion above we interpret the first PCA component dimension as spreadsheet player dimension. In particular, we used the knowledge management model components glyph, data, information, and knowledge (Fig. 4) as scale for this axis. The exact location of these on the x-axis of Fig. 5 was determined by observing the specific transformation function of the spreadsheet player’s information objects in terms of the model (see discussion above).

Now, if we reconsider that a spreadsheet player is commonly considered as an interface for input data, we can clearly see that there are feature groups on the way from non-interpretable glyph to desirable knowledge e.g. as foundation for financial decisions based on the data.

Another aspect under which information objects are perceived is given by the second principal component construct “Represented Data — Implicit Knowledge”. This categorial construct deals with the spreadsheet as a document, with which available knowledge is transformed into distributable data. According to this axis the elements in Fig.3 are spread as *information objects of the document*, because the elements are rated with respect to their communication capacities. For instance, Tables are the most ex-

pressive in terms of representing data, whereas the elements of the meta cluster \mathcal{M} are the most expressive in representing implicit knowledge.

The perceived distinction between spreadsheet player and spreadsheet document is even more remarkable as clarification discussions with some of the interviewees revealed that this distinction was not an explicit one. Prompted to differentiate between the two, the subjects were surprised and not able to distinguish the concepts continuously.

Spreadsheet Author versus Spreadsheet Reader In Fig. 5 we took the element clusters from Fig. 2 into account and stressed their clustering according to the second principal component PC_2 , that refers to information objects of the document. Now we can interpret the distribution of information objects as document features as a *communication timeline* starting from retrieving available knowledge up to compiling it into a *message*. Note that the individual clusters thus show an ordering. Depending on what the document author wants to stress, she makes use of the offered information objects. From this we can cautiously assume that the distinction between a spreadsheet’s author and a spreadsheet’s reader is also perceived. To express the author’s intention the degrees of liberty are highest for the elements of the visual cluster \mathcal{V} , then follows the local cluster \mathcal{L} and are lowest for the meta cluster \mathcal{M} . A reader on the other hand can abstract from the author’s design by using more and more information features along the data interface axis (Fig. 5). We suspect the underlying reason for the controversial location of `Headers` and `Title` in Fig. 3 with respect to this differentiation to be that the content of headers is implicitly already specified by the choice of data to be shown, whereas the title is more general and can thus be more freely chosen by the author.

As the document view is strongly correlated with a spreadsheet author’s view, and the player view correlates with the reader’s perspective, we can reformulate the dichotomy of “document versus player” as “author versus reader”. This gives us an interesting second view on the phenomena involved.

Local Objects versus Global Objects in Spreadsheets As we have seen above our subjects perceived the spreadsheet application objects as different from the `SACHS` information objects. But it is noticeable that `sx:Functional Block` and especially `sx:Localized Info` are close to the separating line in Fig. 3. Only these elements together with `(sx:)Color Coding` (which can also be identified as an `MS Excel` information object) are also local to where the (user-)action is in the application, and by that perceived to belong to the application. Even though `sx:Dependency Graph`, `sx:Relational Arrows`, and `sx:Concept Nodes` are also triggered by local actions of the user, they are far more distant. This is a clear indication that this attribute makes a difference for a spreadsheet user. Together with the dichotomy of “author versus reader” above, we can interpret that the position of an information object relative to the spreadsheet is relevant to readers.

4 Conclusion

In this paper we presented a repertory grid study that investigates properties of information objects in spreadsheets in order to better understand the information qualities given by the SACHS extension for human-spreadsheet interaction. We found four different evaluation schemes for information objects in spreadsheets (see Fig. 6).

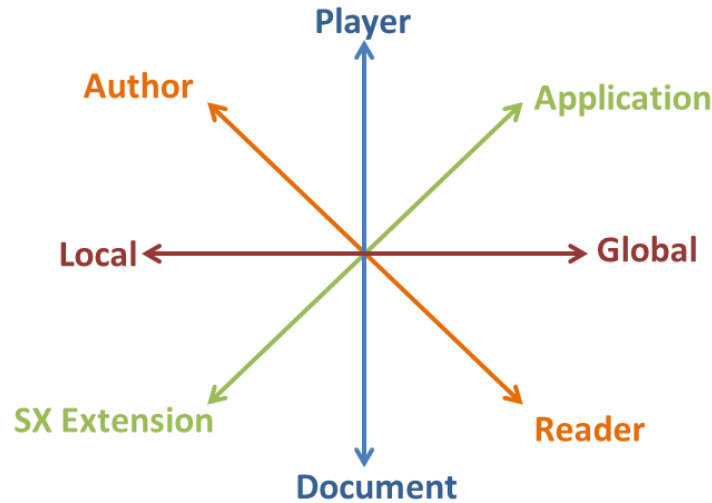


Fig. 6. Quality Dimensions of Information Objects (in Spreadsheets)

The strongest quality dimension is the distinction between spreadsheet player and spreadsheet document. In particular, information is experienced differently, when associated with the spreadsheet player or with the spreadsheet document. A good example is the use of `Formulae`: as a player object the resulting values are considered factual data, as a document object these values are considered to contain implicit knowledge. We conjecture that the distinction between a document player and its document *is* strongly perceived by users in general, even though this study shows this only for spreadsheet applications. If so, then this has extensive consequences for future design options. For instance, we can distinguish between a presentation player like MS PowerPoint and a *.ppt file. Most services though support only document-independent features and miss out on the document-dependent opportunities. “Grouping of shapes” for example is a nice feature to ease copying or moving of a set of shapes. The fact that they “belong together” is only appreciated on the object level, whereas it very often targets a meta level as well: their semantics.

Another extracted dimension is the distinction between information objects in common spreadsheet applications and ones in the semantic spreadsheet extension SACHS. This is especially pleasing, as we conclude that such additional semantic services are

new wrt. common spreadsheet features. In particular, the analysis of the RGI shows that features of the semantic extension do not duplicate already existing features of spreadsheet players. They are perceived by users as additional services targeting implicit general information provided by the player and specific information anchored in the document. They are conceived as features that transform information into knowledge. This doesn't prove the usability of SACHS itself, but it opens a new area of spreadsheet user requirements. Note that this can be generalized beyond spreadsheets as well. For instance, the `sx:Dependency Graph` can also be used for lectures in a presentation application like OO Impress or in a CAD application like Autodesk Inventor.

The distinction of local and non-local positioning of information objects is also of consequence for general application extensions. For instance, this is made use of in [3].

Last, the perceived differentiation of spreadsheet users into authors and readers allows a much better fine-tuning of services. Even though the existence of both groups has been recognized, the interface design for players has not yet seriously taken this distinction into account. A *.ppt document e.g. serves divergent needs of a lecturer versus a student, but basically only former are paid attention to.

All in all, this work has shown that semantic extensions of spreadsheets are perceived as offering distinct functionalities, that turn spreadsheets from data interfaces to knowledge interfaces.

Acknowledgement This work has been funded by the German Research Council under grant KO-2484-10-1.

References

- [1] John Seely Brown and Paul Duguid. *The Social Life of Information*. Harvard Business School Press, 2000.
- [2] Thomas H. Davenport and Laurence Prusak. *Working Knowledge*. 2000th ed. Harvard Business School Press, 1998.
- [3] Catalin David et al. "Semantic Alliance: A Framework for Semantic Allies". In: *Intelligent Computer Mathematics*. Ed. by Johan Jeuring et al. Vol. 7362. Lecture Notes in Computer Science. Springer Berlin Heidelberg, 2012, pp. 49–64. ISBN: 978-3-642-31373-8. DOI: 10.1007/978-3-642-31374-5_4. URL: http://dx.doi.org/10.1007/978-3-642-31374-5_4.
- [4] Ari Ginsberg. "CONSTRUING THE BUSINESS PORTFOLIO: A COGNITIVE MODEL OF DIVERSIFICATION[1]". In: *Journal of Management Studies* 26.4 (1989), pp. 417–438. ISSN: 1467-6486.
- [5] J. Gower. "Generalized procrustes analysis". In: *Psychometrika* 40 (1 1975), pp. 33–51. ISSN: 0033-3123.
- [6] James W. Grice. *Generalized Procrustes Analysis Example with Annotation*. Manuscript at http://psychology.okstate.edu/faculty/jgrice/personalitylab/GPA_Idiogrid_Example.pdf. 2007.

- [7] Marc Hassenzahl and Rainer Wessler. “Capturing Design Space From a User Perspective: The Repertory Grid Technique Revisited”. In: *International Journal of Human-Computer Interaction*. 3rd ser. 12 (2000), pp. 441–459. ISSN: 1044-7318. URL: http://www.informaworld.com/10.1207/S15327590IJHC1203&4_13.
- [8] Stephanie Heidecker and Marc Hassenzahl. “Eine gruppenspezifische Repertory Grid Analyse der wahrgenommenen Attraktivität von Universitätswebsites”. In: *Mensch & Computer*. Ed. by Tom Gross. Oldenbourg Verlag, 2007, pp. 129–138.
- [9] Morten Hertzum and Torkil Clemmensen. “How do usability professionals construe usability?” In: *Int. J. Hum.-Comput. Stud.* 70.1 (2012), pp. 26–42.
- [10] Devi Jankowicz. *The Easy Guide to Repertory Grids*. Wiley, 2003. ISBN: 0470854049.
- [11] George Kelly. “International Handbook of Personal Construct Technology”. In: John Wiley & Sons, 2003. Chap. A Brief Introduction to Personal Construct Theory, pp. 3–20.
- [12] Andrea Kohlhase. “Towards User Assistance for Documents via Interactional Semantic Technology”. In: *KI 2010: Advances in Artificial Intelligence*. Ed. by Rüdiger Dillmann et al. LNAI 6359. Karlsruhe, Germany, 2010, pp. 107–115.
- [13] Andrea Kohlhase and Michael Kohlhase. “Compensating the Computational Bias of Spreadsheets with MKM Techniques”. In: *MKM/Calculemus Proceedings*. Ed. by Jacques Carette et al. LNAI 5625. Springer Verlag, July 2009, pp. 357–372. ISBN: 978-3-642-02613-3. URL: <http://kwarc.info/kohlhase/papers/mkm09-sachs.pdf>.
- [14] Merriam-Webster. *documentz* — Merriam-Webster. [Online; accessed on 2012-09-18]. 2012. URL: <http://www.merriam-webster.com/dictionary/document>}}.
- [15] Raymond R. Panko. “Spreadsheet Errors: What We Know. What We Think We Can Do.” In: *Symp. of the European Spreadsheet Risks Interest Group (EuSpRIG 2000)*. 2000.
- [16] Stephen G. Powell, Kenneth R. Baker, and Barry Lawson. “A critical review of the literature on spreadsheet errors”. In: *Decision Support Systems* 46.1 (2008), pp. 128–138.
- [17] Stephen G. Powell, Barry Lawson, and Kenneth R. Baker. “Impact of Errors in Operational Spreadsheets”. In: *CoRR* abs/0801.0715 (2008).
- [18] G. Probst, St. Raub, and Kai Romhardt. *Wissen managen*. 4 (2003). Gabler Verlag, 1997.
- [19] Felix B. Tan and M. Gordon Hunter. “The Repertory Grid Technique: A Method for the Study of Cognition in Information Systems”. English. In: *MIS Quarterly* 26.1 (2002), pp. 39–57. ISSN: 02767783.
- [20] Terry Winograd. “The Spreadsheet”. In: *Bringing Design to Software*. Ed. by Terry Winograd et al. Addison-Wesley, 2006, pp. 228–231.
- [21] Hsin-Hung Wu and Jiunn-I Shieh. “Applying repertory grids technique for knowledge elicitation in quality function deployment”. In: *Quality Quantity* 44 (6 2010), pp. 1139–1149. ISSN: 0033-5177.