Wagner, Ina; and Kjeld Schmidt (2024): The work to make coordination technologies work. In: *Proceedings of the 22nd European Conference on Computer-Supported Cooperative Work: The International Venue on Practice-centered Computing on the Design of Cooperation Technologies: Exploratory papers (ISSN 2510-2591), DOI: 10.48340/ecscw2024_ep11*

The work to make coordination technologies work

Ina Wagner* and Kjeld Schmidt*[§]
*University of Siegen, Germany
[§]Copenhagen Business School, Denmark
ina.wagner@tuwien.ac.at; schmidt@cscw.dk

Abstract. Motivated by the long-standing concern in CSCW with the state of digital *coordination technologies* and the ensuing accumulated empirical evidence of how the shortcomings of coordination technologies are handled in practice, this paper presents four examples of coordination technologies and coordinative artifacts that show how workers cope with their shortcomings through workarounds and hacks: CAD systems in architectural practice; the medical record in a cardiology clinic and the problem of ICD data; the IMDS database in the car industry; and the problems of making MRP systems work for the purpose of local planning. Concluding with the question what is required to support workers in their cooperative effort to 'make coordination systems work', the notion of computer support for '*peer-to-peer plan management*' is introduced.

> '"The work to make the network work" is considerable.'
> John Bowers (1994, p. 296)

This study is motivated by the long-standing concern in CSCW with the state of digital *coordination technologies* and the ensuing accumulated empirical evidence of how the shortcomings of coordination technologies are handled in practice. The occasion to revisit this issue arose when looking back on findings from ethnographic and similar in-depth workplace studies that we and our colleagues have been conducting over a period spanning about three decades, in work domains as different as manufacturing planning and control, software engineering, air-traffic control, automobile design, architectural design, oncological treatment, and cardiological treatment. While our more immediate goal is a comparative analysis of coordinative artifacts in these domains, we were again struck by how coordination technologies often offer insufficient or no support to workers in handling alignment problems that occur in their daily work and how workers nonetheless *make them work*. The aim of this paper is, first, to point to, or remind us of, these problems and the often innovative ways in which workers cope with them through workarounds and hacks, and second, to outline, or resurrect, a research program aiming at developing computer support for the work of *making coordination technologies work*.

The term ‘coordination technologies’ covers a broad and manifold category of software applications used in contemporary workplaces. Initially developed in the 1950s, this category of applications has developed significantly since then and now encompasses systems as diverse as airline reservation systems, air traffic control systems, computer aided design (CAD) systems, facilities management systems, manufacturing resource planning (MRP II) or enterprise resource management (ERP) systems, software configuration management systems, patient record systems, accounting systems, inventory management systems, project management systems, group calendar systems, and much more. What the instances of this category of technology have in common are, basically, two features. First, they facilitate and regulate the distributed but interdependent work activities of multiple workers engaged in specialized local activities. Or in other words, they serve as coordinative infrastructures to these settings. Second, they are constructed on and around a distinct computational model of the interdependencies of the work arrangement in question, which serves as a calculus in some essential respects. A coordination system can best be understood as a family of coordinative artifacts, typically predicated on received (paper-based) techniques, which have now been integrated computationally, such as, e.g., a group calendar system in which the established techniques and concepts of calendar and timepiece form the foundation of a complex of techniques and concepts such as meeting, deadline, invitee, appointment, meeting room, etc.

However, while indispensable, the digital coordination systems in which these technologies are embodied at the same time pose significant problems to practitioners: their practical integration in the situated flow of work is hampered by the simple fact that these technologies have not been designed to be integrated in actual work practices. Work organizations and coordinative practices have rather been redesigned, or truth be told: *painfully adapted*, to these systems.

Several problems conspire to hamper the development and deployment of digital coordination technologies. Coordination technologies are, first of all, as all artifacts that embody a ‘plan’ or ‘model’ of the world in turn is to be, ‘local and temporary closures’ (Gerson and Star, 1986, p. 263). That is to say, their application is inexorably contingent and requires skill. For a given computational model, however well designed and however ‘intelligent’, ‘situations unanticipated by the original system designers will inevitably arise’, ‘for both practical and theoretical reasons’, and situations will always be possible where the system ‘is beyond its bounds’ (Roth and Woods, 1989, p. 237). Digital coordination technologies today are not designed so as to support users in dealing with contingencies. They offer no or little in terms of which users can meet contingencies. Workers instead strive to make them work by engaging in all kinds of workarounds and plain hacks.

On the other hand, coordination technologies are based on a single conceptual structure or schema, or in some cases, such as MRP or ERP systems, a few closely related conceptual structures. Coordination technologies of course share this limitation with the coordinative artifacts that have come down to us. They all typically embody a single conceptual structure. However, the conventional (paper-based) coordinative practices and artifacts we are talking about here typically comprise a multitude of artifacts used in conjunction. In the development of (computational) coordination technologies only a select few of these artifacts were emulated and integrated in the system, and they rarely, if ever, incorporate the entire received repertoire of coordinative artifacts of the given practice. In the development of coordination technologies, coordinative practices were subjected to the Procrustes Bed of managerialist software engineering.

That is, coordination technologies are carefully devised to address limited and specific issues of coordination, initially often also issues of concern only to specific work domains. As a consequence, they are reciprocally closed. Architects or engineers using a CAD application, for example, are bound to find that it does not interface with the calendar system, the project management system, and the document or workflow management system they are likely to also use. Nor can the functionalities of coordination technologies be integrated with other work tools (word processors, desktop publishing applications, process control systems). These limitations are fundamental and cannot be overcome simply by devising a few *ad hoc* ‘application program interfaces’ (APIs). The number of possible combinations grows exponentially with the number of different systems to be ‘interfaced’ and defeats such a strategy.

‘A world of pain’

The examples we briefly present in the following[1] concern different types of coordinative technologies; hence, they pose different problems for complex distributed work. The CAD system with its layer organization, designed to align different plan drawings requires an enormous amount of manual work to manage task interdependencies. It also offers no support with respect to designing elements of different scales. The second case is a medical record system in a cardiology clinic that shows signs of falling apart, due to the fact that different medical teams act as custodians of an individual patient’s illness trajectory. The IMDS database in the car industry brings unresolved problems of mapping different part numbering systems in use in the big network of care manufacturers and suppliers to the fore. Finally, MRP systems are not adapted to local use, with the consequence that workers need to create their own coordinative artifacts (and copies of them).

CAD systems in architectural practice

CAD plans date back to late 1950s and early 1960s. The first stand-alone commercial CADD platform IDIIOM was developed during the mid-1960s at Information Displays, Inc. (IDI). It was a general-purpose design workstation, and the first software drafting package (Bissell, 1998). Important precursors to CAD technology were the Automatic Programming Tool (APT) developed by Ross at MIT in the 1950s to control machines numerically (Ross, 1984) The movement in the late 1970s from direct numerical control (DNC) to computer numerical control (CNC) was a breakthrough for CAD/CAM (O’Connell, 1987). Another key to CAD/CAM's development was computer graphics, which accommodated both design and manufacturing activities (Machover, 2002). Kale and Arditi (2005) distinguish three generations of CAD technology: computer aided drafting, geometric modelling, and product modelling. Most architectural offices use AutoCAD, a desktop software application for 2D and 3D design and drafting which was first launched in 1982.

First of all, AutoCAD does not support architects and other building specialists in drawing concurrently on different layers of one and the same plan. It also does not support the development and maintenance of conventions. When 20 or more people work in a project, this results in considerable coordination problems. The management of task interdependencies in a CAD system is supported indirectly by its layer organization, a key feature. It is based on overlay drafting that was used by architects well before the adoption of CAD systems. Layering is a method of aligning related plan drawings. This is done by dividing a CAD drawing into

---

[1] The examples are selected from the authors’ corpus of fieldwork (e.g., Carstensen, Schmidt and Wiil, 1999; Odgaard et al., 1999; Carstensen et al., 2001; Carstensen and Schmidt, 2002; Schmidt and Wagner, 2004; Jacucci, Tellioğlu and Wagner, 2007; Bansler et al., 2011; Bansler et al., 2016).

multiple levels or categories, each with its own name and attributes. First, a base drawing is prepared, generally the floor plan. The related drawings are prepared directly over the base and aligned. This ensures that all structural features are aligned as well as all overlapping electrical, piping and other facilities (Hepler, Wallach and Hepler, 2012).

Layering affords the division of labor as it helps maintain the fit between the contributions of the different external specialists to the detailed planning of a building. All information pertaining to a particular task and purpose, such as fire protection, escape routes, ventilation system or structural elements, can be extracted from the file by copying the relevant set of layers. A first limitation, however, is that this process involves substantial manual work: the construction engineer or the heating and ventilation specialists will for instance receive a copy of the relevant layers from the central CAD plan, work on them, and return them to the architects for re-integration into the system. Much of this work is done by copying and pasting.

A second issue results from the limitation of scale. A large building contains hundreds of details, which can either be left open, to be decided upon later by the construction company or craftspeople, or they can be carefully designed up front. However, detail drawings are of a scale of 1:20, 1:10, 1:5 or even 1:1 and therefore cannot be fitted into the central CAD construction drawing. A common solution to this problem is to reference detail drawings in the CAD plan by a simple symbol such as 'circle' and the detail number, a numerical code (Figure 1). This limitation does not only apply to detail drawings but also to components (referenced by 'rectangle' and component ID) and product specifications, all elements that cannot be represented directly in the CAD plan (Schmidt and Wagner, 2004).

Figure 1. References to detail drawings (in circles) in a CAD plan.

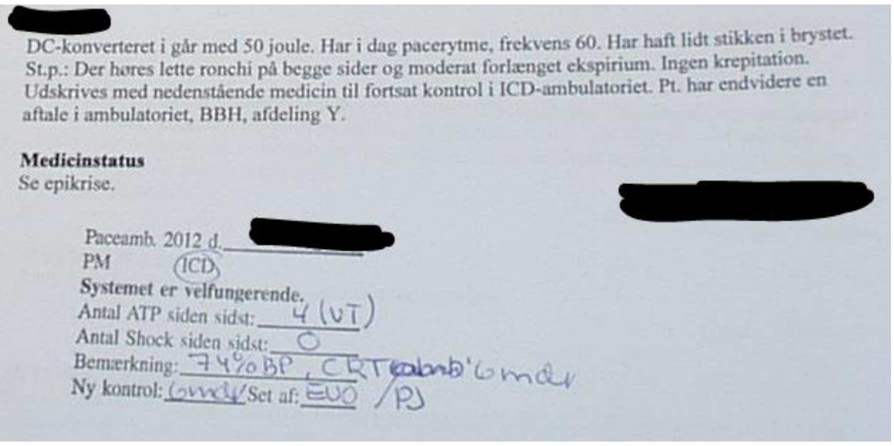
CAD technology supports the coordination of distributed work - the multiple interdependencies that need to be mastered, with each design change potentially propagating through a building – but only in indirect ways. Aligning the contributions of different specialists requires considerable manual work; so does aligning representations of elements of a building on different scale. Moreover, there is no concept of workflow: what needs to be done next, by whom, where, how – the questions that Strauss (1985) sees as fundamental to the alignment of distributed activities.

The medical record and the problem of ICD data

The second example is based on a study of a cardiology clinic and, more specifically, how patients with ICD implants are monitored on a regular basis, both in terms of ordinary cardiology and in terms of the functioning of the device (Bansler et al., 2016). This is done by different clinical workers at different locations: on one hand by the cardiologist at the patient's regional hospital and by the patient's own doctor (GP), and on the other by the ICD cardiologists, or 'electrical doctors' as they are called at the cardiology clinic and by the bio-technicians at the hospital's out-patient clinic who download and print data from the implanted device for specialist scrutiny. Newer ICD devices can be scanned remotely, while the patient is at home (via wireless download to a reader and subsequent transmission to the hospital over the Internet). The data then have to be recorded, filtered, interpreted, classified, put on record, handed over, etc.

Coordination of activities related to ICD devices currently occurs through various artifacts and communication methods used in conjunction. The key artifact is the heart center's patient folder. It holds up to about 500 sheets of paper, some loose sheets, some stapled together. The patient folder is hefty simply because it delineates the trajectory of chronically ill patients as represented in notes, lab reports, test results, clinical imagery, and so on that have accumulated over time, typically several years, and it thus gains size and weight over time.

The medical record concerning a particular patient is not confined to the content of the patient folder. In fact, the clinical record is distributed over an assortment of paper-based and electronic archives and databases. In this context, the most important 'satellite record' is what is called the 'green folders'. It is a large set of suspension folders, housed in about 40 cabinet drawers in the hallway. Each folder contains information about a particular patient's ICD unit, its configuration, and printouts from the data accumulated by the unit. It consists of documents related to the individual unit's status (battery level) as well as its interventions (date, time) since the last reading, along with graphs. The 'green folders' are kept by the bio-technicians at the outpatient clinic and are only occasionally accessed by doctors.



[redacted]

DC-konverteret i går med 50 joule. Har i dag pacerytme, frekvens 60. Har haft lidt stikken i brystet. St.p.: Der høres lette ronchi på begge sider og moderat forlænget ekspirium. Ingen krepitation. Udskrives med nedenstående medicin til fortsat kontrol i ICD-ambulatoriet. Pt. har endvidere en aftale i ambulatoriet, BBH, afdeling Y.

Medicinstatus
Se epikrise.

[redacted]

Paceamb. 2012 d. [redacted]
PM [✓ ICD]
Systemet er velfungerende.
Antal ATP siden sidst: 4 (VT)
Antal Shock siden sidst: 0
Bemærkning: 74%BP, CRT [illegible]' 6 mdr
Ny kontrol: 6 mdr Set af: EVO /PJ

Figure 2. A summary of the ‘read-out’ of an ICD (number of events, etc.) is manually transferred to the clinical notes in the patient folder, using a stamp with empty fields.

Hence, there are two medical record systems that are maintained by different specialists. To counteract the fragmentation of the medical record and ensure a minimum of coordination across the two record systems, a summary of the ‘read-out’ (number of events, etc.) is manually entered into the clinical notes in the patient folder (Figure 2).

The IMDS database in the car industry

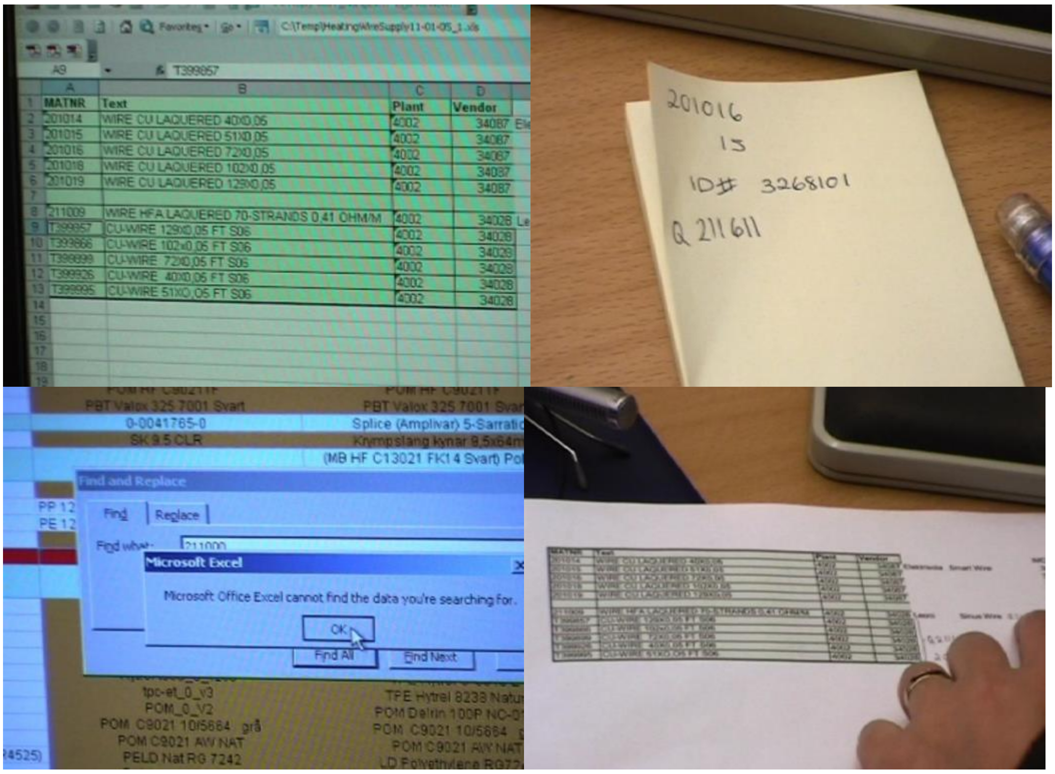
Part numbers are key identifiers in product development and manufacturing. Part numbering systems allow easy recognition of a part through its significant aspects. An ‘intelligent’ part number embeds important information about, e.g., usage, function, material properties, color, finish, mounting or assembly interface of a specific part. Although different manufacturing domains have developed standards and classification systems to resort to, as well as some general rules about how to compose a part number, its structure very much depends on the specifics of a part and its production process. Part numbering is extremely important for a supply chain to be successful. This is why it should be usable in the whole industry. However, large companies tend to use their own part numbers, and mapping between systems is not sufficiently supported.

A study at *Carparts,* a 1st (in some cases) and 2nd (in other cases) tier supplier of the Automotive Supply Chain, brought some of the difficulties of mapping different part numbering systems to the fore (Jacucci, Tellioğlu and Wagner, 2007; Schmidt, Tellioğlu and Wagner, 2009). The particular occasion was the introduction of the *International Material Data System (IMDS)* in the automotive industry in compliance with the ‘End-of-Life Vehicle directive’ issued by the EU in 2000 upon an initiative of German car manufacturers. This system is supposed to track chemical ingredients of parts and assemblies across the entire automotive Original Equipment Manufacturer (OEM) supply chain. It is:

> '[…] designed to act as an easily accessible database to help manufacturers record and track material usage. The system supports recyclability and recoverability of materials in a vehicle and addresses the disposal of substances of concern' (HP International Material Data System - Fact Sheet - 4AA4-0326ENW).

The IMDS was a joint development of Audi, BMW, Daimler, EDS, Ford, Opel, Porsche, VW, and Volvo. Further manufacturers have since joined, and IMDS has become a global standard used by almost all global OEMs. The IMDS database supports searching for materials and suppliers, as well as creating materials and components. The latter requires inputting all substances, their percentage and weight, and, in the case of some substances (e.g., with a polymer classification) to specify how they are used/the parts are made, as well as answer 'recyclate questions'.

Databases are special kinds of coordinative artifacts. In the case of the IMDS, which is used by all actors in the supply chain, it allows the tracing of hazardous substances back to the individual part and work with suppliers to reduce, control, or eliminate the hazard. This presupposes a standardized terminology and requires specialized chemical knowledge. However, there are many unresolved issues around the IMDS database.



MATNR	Text	Plant	Vendor
201014	WIRE CU LAQUERED 40X0,05	4002	34087
201015	WIRE CU LAQUERED 51X0,05	4002	34087
201016	WIRE CU LAQUERED 72X0,05	4002	34087
201018	WIRE CU LAQUERED 102X0,05	4002	34087
201019	WIRE CU LAQUERED 129X0,05	4002	34087
211009	WIRE HFA LAQUERED 70-STRANDS 0,41 OHM/M	4002	34028
T399857	CU-WIRE 129X0,05 FT S06	4002	34028
T399866	CU-WIRE 102x0,05 FT S06	4002	34028
T399899	CU-WIRE 72X0,05 FT S06	4002	34028
T399926	CU-WIRE 40X0,05 FT S06	4002	34028
T399995	CU-WIRE 51X0,05 FT S06	4002	34028

201016
15
ID# 3268101
Q 211611

Microsoft Excel

Microsoft Office Excel cannot find the data you're searching for.

Figure 3. Trying to enter information about part SMART TCU-UN from a supplier and to add it to the IMDS database using a customer part ID.

Placed at *Carparts* within a network of in-house specialists, suppliers and customers, workers have to face the complexities of making IMDS work in practice.

For example, when one part is delivered directly to the car producer and another one to let's say the seat producer, each of those companies uses a different part number. Suppliers, in particular small ones, have problems in providing detailed information about their parts that are required by the IMDS system. The responsible person at *Carparts* has problems to retrieve this information in order to be able to add it to the IMDS database. She struggles with numerous documents while sending out an email, making phone calls, printing out lists, carrying them to another office to discuss the problem, and so forth. She consults an MS Excel sheet with part numbers of different heating wires, finding out that it has not been updated, goes through an annotated printout of a list with part numbers, and jots down alternative part numbers (Figure 3).

The fact that the mapping problems between different part numbering systems have not been resolved, affects many workplaces at *Carparts*, among them testing, purchase, and sales.

Making MRP systems work for the purpose of local planning

The fourth example is based on a series of six studies in Danish manufacturing plants all engaged in devolution of operational control to locally autonomous working groups: a factory producing facilities for the distribution of electrical power from power plants to consumers, a medium-size enterprise producing steel wardrobes and steel wire products, one of the largest shipyards in Europe, a leading European power cable manufacturer, a leading manufacturer of propulsion plants for smaller and medium-sized ships, and a leading manufacturer of sound and vibration measurement equipment. (For summaries of findings, see Odgaard, 1994, 1995; Carstensen, Schmidt and Wiil, 1999; Odgaard et al., 1999; Carstensen et al., 2001; Carstensen and Schmidt, 2002).

Cooperative work in manufacturing is basically and generally characterized by the challenge of coping with a very high level of coordinative complexity. For each product a very large number of parts and subassemblies are to be produced in different predetermined sequences at specialized workstations, and in contemporary plants multiple products are being produced in parallel, competing for the same resources. To deal with this, engineers have over the last century or so developed a sophisticated set of coordinative techniques and associated artifacts. Central to these is the 'bill of materials', a hierarchical representation of the total set of parts (components and subassemblies) comprising the final product. Taking that representation as a framework and combining it with the empirically determined times to produce each part (preparation, time to obtain materials, set-up time at the workstation, processing, cleaning, etc.) and intermediary transportation between workstations, the planners calculate a 'master schedule' indicating the time for each of the constituent processes to begin to enable the overall production series to meet a given deadline.

Until the emergence of host computers in the 1950, the calculations were entirely manfully performed to calculate a master schedule, a massive task using type- or handwritten lists of parts, blueprints, etc. Unsurprisingly, with computing power available, this massive task was one of the first to be computerized.

Now, the concept underpinning MRP-based production planning and control addresses planning at the population level, so to speak, that is, at the level of the entire operation (by product or by the portfolio of products to be produced in parallel), not the level of the individual workstation (or group of workstations). And it is at that level that contingencies rule the day: defective tools and machinery, absent workers, defective parts and materials, delays in delivery, etc.

It may, to take an example from a marine diesel-engine manufacturer, turn out that the test run of an engine has to be postponed because the engine block turns out to have fractures. A replacement will have to be ordered and produced, of course, but it also has to be transported across Europe on a truck (even an ‘oversize load’ transportation) which will take weeks. In the meantime, the upstream production of parts and assemblies for this particular engine proceeds as planned by the SAP MRP system, while other production orders competing for some of the same resources (workstations) dutifully wait for this already released production order has cleared the intersection. Or rather, somebody on the shopfloor realized the traffic jam and a new master schedule was generated and the waiting orders were released while the frame made its slow way from Romania to Denmark.

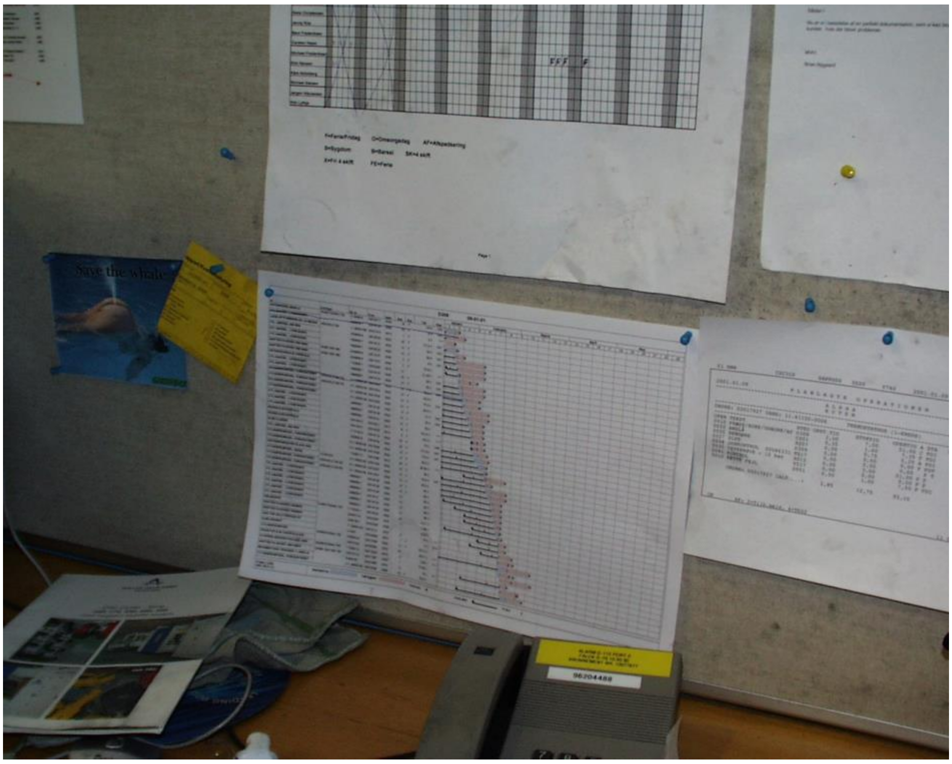


Figure 4. From the ’office’ of the team planner, parts production, MAN B&W Diesel. The Gantt chart at the bottom of the posting board is a print-out of the local production plan.

In fact, the ship engine manufacturer had been experimenting with systematic devolution of operational control to locally autonomous teams on the shopfloor and, realizing that the MRP system was too unwieldy for local planning purposes, the production engineers and planners had developed a hack to extract data from SAP into the Microsoft Project application so that the planners of the local teams could rearrange jobs (order of priority, timing) within their remit as they saw fit. However, this hack did not help in the case of the defective block because it was based on a data master schedule which had not been updated to reflect the upstream consequences of a postponed test and the hack did not support the propagation of such information upstream in the chain. Only gossip and shoptalk among workers made it transpire.

Implications for design: ‘Peer-to-peer plan management’

The complications with making plans work that we have just sketched have quite different causes: In the case of architectural practice, the need to make plans at different scales of representation makes it necessary to operate with different sets of plans: detail plans, components, and the central CAD plan with its scores of layers. To keep order in the vast distributed repository, architects, among many other measures, insert a snippet of text (a comment) next to the relevant object in the layer informing a reader of the existence and ID of a detail drawing. The two representations are thereby cross-indexed, but the textual indicator is as dead as a tombstone, its meaning precariously predicated on proximity in a chart, and amendments to the CAD layer might easily erase the indicator or render it pointless. This is hardly a rigorous solution.

The indicator pointing from the central patient folder to the corresponding file in the cardiology clinic's satellite archive is caused by the arrival of cardiological technology that automatically generates a large amount of clinical data that, if automatically included in the record, would make the record useless for doctors as a basis for making decisions. The solution in this case is to insert a rudimentary textual summary of the latest readout of the ICD in a form stamped onto the clinical progress notes. But again, it is hardly robust. Nor can it be retained, in the observed form or in the existing division of labor vis-à-vis the clinical record, in the transition to the computer-based clinical record system.

The complications faced by the workers at *Carparts* are quite different from these examples of indexing objects in distributed repositories. It is caused by the fact that standardization has its limits. There may or will at any time be areas outside of the domain under the superimposed schema of the standard. Partly due to the fact that the automotive industry, like any other, is not an island; it has supply chains that overlap with the supply chains of other industries such as construction and agricultural machinery, elevators and escalators, batteries and chips, textiles, and electrical conduits. And partly, received repositories of drawings, etc., may be brought together in a process of business mergers and acquisitions, while reclassification would be uneconomical; likewise novel products, materials, processes, etc. may not as yet have been domesticated under the classification schema. There will inevitably be boundaries at which complications must be handled.

And finally, in the case of the diesel engine manufacturer, the complications described here are caused by the inherent limitations of MRP systems in that they are designed to serve overall planning of the operation as a whole, and cannot rationally be used to deal with local contingencies (defective or delayed parts, etc.) for the simple reason that a master schedule that is changed on a daily or hourly basis is not a master schedule. The limitation here is also one of scale, not spatial but temporal. The team planners' work is therefore to reschedule local tasks as needed and, if necessary, stay in touch and negotiate with their colleagues on other teams in case the effects of contingent deviations from the master schedule spill over from one team to the next.

What is to be done?

What is required to support workers in their cooperative effort to 'make coordination systems work' could be termed computer support for '*peer-to-peer* *plan management*'. By that we mean a family of technology that supports the residual coordinative tasks that fall outside of the remit of a coordination system, taking as input the plan generated by the coordination system (CAD plan, car model, patient records, design issue list, MRP master schedule). It uses the conceptual structure of the coordination system to support the horizontal coordination among distributed workers in managing the plan generated by it: using *the plan as a resource for situated action*, to use Suchman's terminology (1987).

Software developers have of course recognized the challenge of making infrastructural coordination systems work in practice, in real cooperative work settings. But they have generally done so simply by offering additional communication channels: video and audio, e.g., Teams or Zoom (or text-based chat as a substitute), supplementing the coordinative functions of the coordination system. (In fact, the declared design aim typically is not to reduce the complexity of shopfloor coordination but rather to facilitate remote collaboration, thus adopting the approach that has dominated CSCW for decades).

What is crucial, however, for collaborative technologies to make significant progress in this regard is that 'the work to make plans work' is done *in domain specific terms*. What is required is digital coordination support in terms of the categories of the domain as expressed in computational terms: objects and object classes (CAD plan objects, cardiological states, MRP object classes such as parts, processes, and resources as well as timeline points).

For example, consider a peer-to-peer plan management system (P3MS) for architects working with a CAD plan of a building. For each 'move' considered by our architect (moving a wall, say), a number of contingencies may arise. The architect may want to explore the history of the design process with respect to this object or make an annotation raising objections to this particular design decision, embedding the note as a property of the object (as an embedded Request for Comments). More dramatically, the considered move may conflict with other design decisions, and the system may detect that by traversing the web of the CAD model of the construction and provide notifications to relevant parties according to a prespecified protocol (perhaps as 'semi-structured messages', perhaps by opening a chat thread, connected to the object).

Or to take another example. In case of local deviations from the master schedule plan generated by the MRP system, workers will need support for checking if the considered plan amendments might be in conflict with tasks assigned to other actors, at other station, both down- and up-stream. If such conflicts are detected by the P3MS, again by traversing the model of interdependencies, the P3MS may be used to notify affected workers at stations down- and up-stream of the intended plan changes, and, if necessary, establish a communication channel for negotiations. And finally, for purposes of accountability, the P3MS should retain a record of local plan deviations, again as properties of the affected objects (jobs, stations, work-in-progress).

These observations are hardly revolutionary revelations. In advanced manufacturing, for example, efforts to develop technologies (such as 'manufacturing execution systems') to support cooperative plan management, taking MRP models as the infrastructure, have been ongoing for many years but seem to be stalling (see, e.g., McKay and Black, 2007; Saenz de Ugarte, Artiba and Pellerin, 2009; Järvenpää et al., 2015; Mantravadi and Møller, 2019).

What would seem to be required is a more radical approach, the practice-centered CSCW approach. Coordinative practices are domain specific (they are, after all, types of situated action) and the challenge is to develop tools that enable domain actors to express the residual tasks arising contingently at the boundary of coordination systems and manage them cooperatively. To move the development of peer-to-peer plan management beyond *ad hoc* solutions requires bottom-up development of programming environments that provide higher-level categories of object classes for coordinative work (such as, for example, 'Who gets to do what, when, and how, how much' (to invoke Strauss, 1985, p. 9), but also what *has* been done, by whom, where, etc. and schemas for naming, classifying, and placing things, etc.). It involves a process of systematic abstraction, 'from the ground up' (Hughes et al., 1994, p. 129). This is what makes it difficult to grasp and do.

In sum, the visionary research program for CSCW that Irene Greif sketched 36 years ago remains highly relevant:

> 'Designers who draw pictures, software developers who jointly write code, financial analysts who collaborate on a budget—they all need coordination capabilities is an integral part of their work tools. That means coordination support within the CAD engineer's graphics package, within the programmer's source-code editor, within the budget writer's spreadsheet program. It means support for managing multiple versions of objects, be they pictures, programs, or spreadsheets. It means ways to distribute parts of the object for work by contributing group members, ways to track the status of those distributed parts, ways to pull completed objects back together again. ' (Greif, 1988, pp. 8 f.).

References

Bansler, Jørgen P.; Erling C. Havn; Troels Mønsted; and Kjeld Schmidt (2011). A study of the fragmentation of the medical record. In P. Bjørn; F. Kensing; and L. R. Christensen (eds): *Infrastructures for Healthcare 2011: Global Healthcare: Proceedings of the 3rd International Workshop, 23-24 June 2011, Copenhagen*. Copenhagen: IT University of Copenhagen, pp. 94-97.

Bansler, Jørgen P.; Erling C. Havn; Kjeld Schmidt; Troels Mønsted; Helen Høgh Petersen; and Jesper Hastrup Svendsen (2016). Cooperative epistemic work in medical practice: An analysis of physicians' clinical notes. *Computer Supported Cooperative Work (CSCW)*. Vol. 25, no. 6, December 2016, pp. 503-546.

Bissell, Don (1998). Was the IDIIOM the first stand-alone CAD platform? *IEEE Annals of the History of Computing*. Vol. 20, no. 2, April-June 1998, pp. 14-19.

Bowers, John M. (1994). The work to make a network work: Studying CSCW in action. In J. B. Smith; F. D. Smith; and T. W. Malone (eds): *CSCW'94: Proceedings of the Conference on Computer-Supported Cooperative Work, 24-26 October 1994, Chapel Hill, North Carolina*. New York: ACM Press, pp. 287–298.

Carstensen, Peter H.; Kjeld Schmidt; and Uffe Kock Wiil (1999). Supporting shop floor intelligence: A CSCW approach to production planning and control in flexible manufacturing. In S. C. Hayne (ed.): *GROUP 1999: International Conference on Supporting Group Work, 14-17 November 1999, Phoenix, Arizona*. New York: ACM Press, pp. 111-120.

Carstensen, Peter H.; Hans Jørgen Lynggaard; Kjeld Schmidt; Alex Skandorff Vestergaard; and Uffe Kock Wiil (2001). *IT-støtte til produktionsgrupper: Debatoplæg fra IDAK-projektet*. København: CO-Industri etc., June 2001.

Carstensen, Peter H.; and Kjeld Schmidt (2002). Self-governing production groups: Towards requirements for IT support. In V. Marík; L. M. Camarinha-Matos; and H. Afsarmanesh (eds): *Knowledge and Technology Integration in Production and Services: Balancing Knowledge and Technology in Product and Service Life Cycle. IFIP TC5/WG5.3. Fifth IFIP International Conference on Information Technology in Manufacturing and Services (BASYS'02), 25-27 September 2002, Cancun, Mexico*. Boston, Mass.: Kluwer Academic Publishers, pp. 49-60.

Gerson, Elihu M.; and Susan Leigh Star (1986). Analyzing due process in the workplace. *ACM Transactions on Office Information Systems*. Vol. 4, no. 3, July 1986, pp. 257-270.

Greif, Irene (1988). Overview. In I. Greif (ed.): *Computer-Supported Cooperative Work: A Book of Readings*. San Mateo, Calif.: Morgan Kaufmann Publishers, pp. 5-12.

Hepler, D.J.; P.R. Wallach; and D. Hepler (2012). *Drafting and Design for Architecture & Construction*Cengage Learning. <https://books.google.at/books?id=KWoJAAAAQBAJ>

Hughes, John A.; Wes W. Sharrock; Tom A. Rodden; Jon O'Brien; Mark Rouncefield; and Dave Calvey (1994). *Field Studies and CSCW (COMIC Deliverable D2.2)*. Lancaster, UK: Computing Department, Lancaster University. (*COMIC, Esprit Basic Research Project 6225*. <https://dl.eusset.eu/handle/20.500.12015/4110>

Jacucci, Gianni; Hilda Tellioğlu; and Ina Wagner (2007). Practices and cultures of knowledge management. In: *CCE'07: 5th Workshop on Challenges in Collaborative Engineering, Kraków, Poland, 11-13 April 2007*. pp. 99-112.

Järvenpää, Eeva; Minna Lanz; Henri Tokola; Tapio Salonen; and Mikko Koho (2015). Production planning and control in Finnish manufacturing companies–Current state and challenges. In C. F. Oduoza (ed.): *25th International Conference on Flexible Automation and Intelligent Manufacturing (FAIM2015), 23-26 June 2015, Wolverhampton, UK*. Gloucester, UK: The Choir Press. <https://urn.fi/URN:NBN:fi:tty-201705191420>

Kale, Sedar; and David Arditi (2005). Diffusion of computer aided design technology in architectural design practice. *Journal of Construction Engineering and Management*. Vol. 131, no. 10, October 2005, pp. 1135-1141.

Machover, Carl (2002). Oral history interview. Interviewed by P. Frana, 20 June 2002, at White Plains, New York. Interview transcript, Charles Babbage Institute, University of Minnesota, Minneapolis (OH 362).

Mantravadi, Soujanya; and Charles Møller (2019). An overview of next-generation Manufacturing Execution Systems: How important is MES for Industry 4.0? *Procedia Manufacturing*. Vol. 30, January 2019, pp. 588-595. <https://www.sciencedirect.com/science/article/pii/S2351978919301155>

McKay, Kenneth N; and Gary W Black (2007). The evolution of a production planning system: A 10-year case study. *Computers in Industry*. Vol. 58, no. 8-9, December 2007, pp. 756-771.

O'Connell, Colette (1987). CAD/CAM (Computer-Aided Design / Computer-Aided Manufacturing): A history of the technology and guide to the literature: Part I. *Science & Technology Libraries*. Vol. 7, no. 4, 1987, pp. 127-154.

Odgaard, Irene (1994). *"Praksis var værre end forventet": Produktionsgrupper på Ålestrup: Grundfos, Denmark. Casestudie-rapport om nye former for arbejdsorganisation*. København: Specialarbejderforbundet i Danmark (SiD), 26 August 1994. – [IMF Project on new forms of work organization (IMF-EMF-FIET.). Unpublished report. In Danish].

Odgaard, Irene (1995). User Requirements for Tools to Support Human Oriented Management of Change. *Advances in Human Factors/Ergonomics*. Vol. 20, no. C, December 1995, pp. 1017-1019.

Odgaard, Irene; Peter H. Carstensen; Kjeld Schmidt; Uffe Kock Wiil; Bruno Clematide; and Maj Fagerberg (1999). *Produktionsgrupper: organisationsudvikling og IT-støtte*. København: CO-industri etc.

Ross, Douglas T. (1984). Oral history interview. Interviewed by W. Aspray, 21 February 1984, at Waltham, Mass. Interview transcript, Charles Babbage Institute, University of Minnesota, Minneapolis (OH 65).

Roth, Emilie M.; and David D. Woods (1989). Cognitive task analysis: An approach to knowledge acquisition for intelligent system design. In G. Guida and C. Tasso (eds): *Topics in Expert System Design. Methodologies and Tools*. Amsterdam: North-Holland, pp. 233-264.

Saenz de Ugarte, B.; A. Artiba; and R. Pellerin (2009). Manufacturing execution system – a literature review. *Production planning and control*. Vol. 20, no. 6, pp. 525-539.

Schmidt, Kjeld; and Ina Wagner (2004). Ordering systems: Coordinative practices and artifacts in architectural design and planning. *Computer Supported Cooperative Work (CSCW)*. Vol. 13, no. 5-6, December 2004, pp. 349-408.

Schmidt, Kjeld; Hilda Tellioğlu; and Ina Wagner (2009). Asking for the moon: Or model-based coordination in distributed design. In I. Wagner, *et al.* (eds): *ECSCW 2009: Proceedings of the 11th European Conference on Computer-Supported Cooperative Work, 7-11 September 2009, Vienna, Austria*. London: Springer, pp. 383-402.

Strauss, Anselm L. (1985). Work and the division of labor. *The Sociological Quarterly*. Vol. 26, no. 1, 1985, pp. 1-19.

Suchman, Lucy A. (1987). *Plans and Situated Actions: The Problem of Human-Machine Communication*. Cambridge: Cambridge University Press.