# What is Social Software?

**Jan van Eijck and Rohit Parikh**

*It is a sunny autumn day, and our protagonists have taken their meals outside, to enjoy the mild rays of the September sun. The NIAS cook Paul Nolte, as always glowing with pride while serving out his delicious food, has prepared a traditional Dutch meal today with sausage, red cabbage and pieces of apple.*

*Computer Scientist:* Hmmm, very tasty. Do you all realize that for the first time NIAS has opened its gates to the likes of us? Logic and computer science used to be outside the compass of NIAS. Moreover, all of our other colleagues are pursuing goals of their own. They can devote themselves exclusively to their individual academic projects, as the NIAS website puts it, and I must say: I envy them. We are the only ones who are supposed to perform a *collective* task. We have to come up with new ideas in an area that hardly exists, but that is supposed to bridge a gap between the humanities and science. A rather tall order, if you ask me.

*Logician:* Yes, but you cannot deny that it is very pleasant here. I enjoyed yesterday evening's concert very much, for instance. One can get used to the ways of NIAS; humanities research is carried out here in a very civilized fashion, indeed. The only thing that worries me right now is the vagueness and vastness of our topic. We are supposed to come up with something we can show after our "Games, Action and Social Software" project here finishes. The trouble is that I have only the vaguest of ideas of what social software actually is or might be.

*Philosopher:* The term "Social Software" was coined by Rohit Parikh, in a paper which appeared in *Synthese* [15]. It had been circulating as a manuscript for some years. Parikh does not give a precise definition but he lists a series of evocative examples, rather in the manner of Wittgenstein in *Philosophical Investigations*. What Parikh has in mind is procedures that structure social

reality, in a very broad sense. He makes a plea for investigating these with the tools of mathematics, logic and computer science. This was taken up by various people. See for instance the PhD thesis of Marc Pauly [16] or that of Eric Pacuit [14].

*Logician:* Now that the term has caught on, I suppose there is little reason for Parikh to come up with a precise definition. Such a definition will cost him the support of people who like his examples but might dislike the way he draws demarcation lines.

*Computer Scientist:* Yes, I think it is wise not to rely too much on Rohit for a definition. In trying to understand what the term "Social Software" might mean, why not take our cue from computer science? Software is what you feed a computer to make it do something useful. Feeding it with appropriate software turns a computer into a text processor, or into a digital entertainment center. As we all know, the dividing line between hardware (the machine) and software (the programs running on the machine) is blurred by the fact that an increasing number of system tasks are carried out by hardware.

*Philosopher:* I suppose that drawing the precise line between hardware and software is not that easy, indeed. But couldn't we agree on the following: what can be changed without changing the machine itself is called software?

*Logician:* Yes, that will do for now. Computer software is roughly divided into *system software*, namely, the software that is needed to make other software run, and *application software*, the software that turns the computer into a tool for a specific task. Taking our lead from computer science, we get the following distinction between social hardware and social software: Social hardware consists of institutions such as schools, churches, law courts, parliaments, banks, newspapers, supermarkets and prisons, while social software consists of the *more specific procedures* followed in these institutions.

*Computer Scientist:* Most computer software is designed, although if you look at large software systems such as the Linux operating system, then these can certainly be viewed as products of evolution of a certain kind. Genetic algorithms are another example. These are search techniques for finding programs for specific tasks by a process of genesis and natural selection, so programs resulting from a genetic algorithm are not designed.

*Philosopher:* There is a large class of social practices that have evolved in the course of development of a civilization. Our practice of eating with knife and

fork while observing certain rules is one of many examples [8; 9]. Other social practices were designed and redesigned over a long period of time, e.g., the principles of common law.

*Computer Scientist:* The division of software in two broad categories carries over to the case of social software too, I suppose. Let us call *social system software* the rules of social interaction that make a society civilized. The rule of law, and the rules of civic behaviour that engender mutual trust among social agents.

*Philosopher:* How did Thomas Hobbes say it? Without social system software our lives would be 'solitary, poor, nasty, brutish, and short.' The theme of *trust* as a quintessential product of social system software has been taken up in our times by Francis Fukuyama [10] and others [6; 20]. No doubt the general principles that constitute aspects of the so-called 'rule of law' [22] would fall under social system software.

*Logician:* What is it you have in mind?

*Philosopher:* Let me give some examples. *Nemo judex in sua causa.* This describes the principle of natural justice that no person can judge a case in which he or she is a party. It seems fairly obvious to us, but then again our societies are partly a product of the Roman law system where this principle evolved. Or take *Nulla poena sine lege*, or *Lex retro non agit.* One cannot be penalised for doing something that is not prohibited by law.

*Logician:* A key principle of law, I suppose, is that nobody shall be judged unheard, which means reasonable opportunity must be given to an accused to defend his side of the case. Without such a principle it is hardly thinkable that a fair jurisprudence could evolve at all.

*Computer Scientist:* Yes, and other principles no doubt have the purpose of ensuring that court cases can terminate. *Ne bis in idem* is an example of this: no legal action can be instituted twice for the same cause of action.

*Philosopher:* Another one, one that I have memorized, is *Volenti non fit injuria.* Someone who knowingly and willingly puts himself in a dangerous situation will be unable to sue for his resulting injuries. Comes in handy quite often as an erudite way of saying 'serves him right'. If you go bungee jumping and get injured, you cannot sue the one who supplied the elastic cord.

*Computer Scientist:* Not everywhere. Some countries require bungee sites to

have liability insurance.

*Logician:* Bungee jumping was just an example, remember. I think we got the point. The main perspective on the law in Dutch society, by the way, seems to be that other basic principle from Roman law: *De minimis non curat lex.* This is taken to mean that the law is not interested in trivial matters. In Dutch society there are many things which are thought not worthy of the law's attention. Possessing less than ten grams of cannabis, for example.

*Philosopher:* I am afraid we are getting side-tracked here. It is obvious that the foundations and principles of legislation are part and parcel of the broad field of social software. But it is not so clear what *we* have to contribute here. I propose we concentrate instead on the social procedures and protocols geared towards specific tasks, such as division of goods, voting, reaching agreement, negotiation, settling disputes, that kind of thing. Let us focus on what one might call *social application software.*

*Computer Scientist:* Fair division of goods is an excellent example. For the fair division between two people we have what in English is called *I cut, you choose.* In Dutch this is called *kiezen of delen.* This is the procedure where one person makes the division, and the other person has the right to choose one of the pieces. Apparently, this is known from antiquity. It appears in a famous medieval story, 'Charlemagne and the Elbegast' [7].

*Philosopher:* A rather peculiar version of this was used by King Solomon in the Old Testament. He took the 'I cut' quite literally, in his proposal to settle a dispute between two women about a baby. He threatened to cut the child in half.

*Computer Scientist:* The case of Solomon and the two women is interesting, for it has been noticed that Solomon's procedure hinges on the surprise element. Suppose Solomon has to settle a second dispute about a child between two women, while his first judgement is well known. Surely, the second pair of women would *both* say that they prefer the other to have the child than for him to die.

*Philosopher:* Yes, the surprise element is crucial for Solomon's procedure to work. If the impostor knows the procedure, she will be able to play strategically, by pretending she is also willing to give up the child. Almost all social procedures are susceptible to strategic behaviour, where it pays not to act according to your real preferences.

*Computer Scientist:* If you ask people to invest real money, you can always force them to reveal their real interests, I suppose. In a second dispute about a child, Solomon would simply propose to sell the baby to the highest bidder, knowing that she had to be the real mother.

*Logician:* Beforehand he should offer them both a generous loan from the Temple funds, to be paid back in monthly installments plus interest. And this time the rules can be publicly announced: bids in closed papyri, highest bidder gets the baby at the offered price, loser pays a fee into the Temple funds to cover court expenses.

*Philosopher:* This might not work if the pretender has more money than the true mother. Better to ask them how many times their annual income they are willing to bid for the child.

*Computer Scientist:* If the bids are in closed papyri, and the first mother offers $A$ times her annual income and the second mother $B$ times her annual income, with $A > B$, then the child should go to the first mother for $B$ times *her* annual income. For this is what she would have paid in an open auction, with the second mother (the 'fake' mother) dropping out at $B$ times annual income.

*Philosopher:* This is called a *sealed bid second price auction*, isn't it? Such auctions are strategy-proof, in the sense that it is never in the interest of the bidders to put in a lower bid than what they believe is the true value.

*Logician:* Yes, such an auction would work in this case. In fact, a variation on this solution was proposed in the literature: see Moore [13] (and also [17]). Suppose the child is worth $A$ times her annual income for the real mother, and $B$ times her annual income for the pretender, with $A > B$. Now the women make their bids in sealed papyri, and Solomon collects the papyri without looking at who handed them in. He announces his procedure to the women. If one of them gives the child to the other, he will consider the case settled. If not, then he will toss a coin to decide who gets the child, and (looking at the bids) rule that that woman will have to pay $M$ times her annual income, with $A > M > B$, and the other woman will have to pay a small fine.

*Philosopher:* Court expenses again.

*Logician:* Yes. Solomon then asks the first woman whether she is willing to give the child to the second woman. If so, all is over and done with. If not, he asks the second woman whether she is willing to give the child to the first

woman. If so, all is over and done with. If not, he tosses his coin, decides who gets the child, and both women pay expenses as stipulated: the woman who gets the child pays $M$ times her annual income, and the other woman pays the fine.

*Computer Scientist:* Ah, I see how this works. If the first woman is not the true mother, she knows she is running the risk of having to pay more than the child is worth to her. She has offered $B$ times her annual income, but if she gets the child she will have to pay more than that, and if she does not get the child she will have to pay a fine. So she will give it up. If the first woman is the true mother, the second woman will know that she is running the risk of ending up with the child at a price she does not want to pay, or ending up with nothing and having to pay a fine. So then *she* will give it up. If both act rationally, the true mother gets the child, at no cost at all. How brilliant!

*Philosopher:* I suppose it is essential that Solomon announces the price $M$ for the winner and the small fine for the loser beforehand. Then both women know that the other one knows what might happen.

*Logician:* Yes, and note that the procedure assumes that the women are both rational, and know of each other that they are rational. If the pretender acts irrationally by refusing to give up the child — 'I will never part with my darling, I just can't, and to hell with the cost' — then she could end up having the child after all.

*Computer Scientist:* The Solomon case is special because the goods are non-divisible. With divisible goods, real money always makes for smoother fair division, I suppose. Here is a procedure for dividing an inheritance between $n$ inheritors: first auction the goods among the $n$ inheritors, next divide the auction revenue in $n$ equal shares.

*Philosopher:* This may not be a fair procedure if some of the inheritors are much poorer than the others.

*Computer Scientist:* OK, but how about the following procedure. This is a simple generalization of *I cut, you choose* to the case of $n$ participants.

> I cut out a piece of the inheritance that I know I am satisfied with and offer it to the others. If someone else wants it, I give it to him, and we continue with $n-1$ players. If no-one else wants it, I take it myself and let the other players continue.

Doesn't this guarantee that everyone gets his fair share? So what's the big deal about cake cutting algorithms?

*Philosopher:* In the literature [4; 3] it is common practice to use cake cutting as a metaphor for a *division of a single heterogeneous good.* Dividing a piece of land at inheritance would be an example. The cake has different toppings that cannot all be cut into pieces with the same composition: it may have turkish delight cherries on top that someone likes but another person abhors, and so on. A cake division is *simply fair* if each of $n$ players feels she received at least $1/n$ of the cake, according to her individual valuation of its parts, that is. I agree that the procedure you propose is simply fair, but your procedure does not rule out the possibility of hard feelings. A cake division is called *envy-free* if each person feels that nobody else received a larger piece. A sure sign of a division being envy-free is that nobody wishes to trade pieces with anyone else. The procedure you propose is not envy-free.

*Computer Scientist:* Ah, I see what you mean. The procedure guarantees that I get what I consider a fair share, but it does not rule out that someone else gets a share that I consider excessive. This explains, by the way, why fair, envy-free division among two is so much simpler than fair, envy-free division among many. If I have received my fair $1/n$ share, I can still be envious because I feel that some of the other $n-1$ pieces are larger than mine. The *I cut, you choose* procedure is fair, and it is envy-free simply because the rest of the cake is a single piece, so there is no possibility for envy.

*Logician:* If the preferences of the players are not the same, then I suppose the typical result of fair division will be that all players feel they have received *more* than their fair share. In fair division there is no objectivity, remember.

*Computer Scientist:* And if the division is also envy-free then each player will feel that she has done at least as well as each of the others. A very satisfactory outcome indeed.

*Philosopher:* Yes, but it is surprisingly difficult to generalize *I cut, you choose.* One of the difficulties, by the way, is that preferences might change while the division is in progress. Consider the case of a land inheritance where you have picked your piece of land. Then the piece of land next to yours has increased in value for me, because of the attractive prospect of having you as my neighbour.

*Computer Scientist:* You are teasing me, but I take your point. But wait,

didn't Rohit's social software paper [15] have a discussion of cake cutting?

*Logician:* Ah, you mean the Banach and Knaster cake cutting algorithm? That is indeed a good example. It goes like this.

> I cut a piece intended for myself. All others consider it. If nobody objects, I get my piece. If someone raises an objection, she has the right to cut off a slice and put that back with the rest of the cake. She then asks if she can have the reduced piece. If nobody objects, she gets it, otherwise someone else takes the knife and reduces the piece a bit further, and so on, until someone gets the trimmed piece. Then on to the next round, with $n-1$ players.

*Computer Scientist:* A nice feature about Parikh's discussion is that he shows how the methods of computer science can be used to argue that the procedure is fair. The key ingredient of the procedure is a loop operation:

> continue to trim the piece until there are no further objections about the size.

If $r$ stands for the action of trimming, and if $F(m,k)$ is the proposition that the main part of the cake is large enough for $k$ people, then we can see that $F(m,k)$ is invariant under the action $r$. If $F(m,k)$ is true before $r$, then it will still be true after $r$ has occurred. Clearly, if one can show that $F(m,k)$ continues to hold through the algorithm, for $k$ running through $n,\ldots,1$, then this establishes that the division is fair, for surely $F(m,n)$ holds at the beginning: the whole cake is large enough for the whole group to begin with.

*Logician:* Yes, and if I remember well, Parikh proposes a game logic to carry out the verification. Don't you think, by the way, that an additional argument would be needed for envy-freeness?

*Philosopher:* Yes, I think you are right. But what I don't like about the algorithm is the way it spoils the cake. You were looking forward to a treat, and you end up with an unappetizing mush of cake, cream and topping.

*Logician:* There is also a version with a continuously moving knife. This leaves the cake intact. See [11].

*Philosopher:* Ah, I take that to mean that we, as social software designers, are allowed to propose improvements on social division procedures. Then how about the following?

> I start by cutting off a piece intended for myself. All others consider it, and are allowed to make money offers on it. If nobody does, I get the piece, without paying for it. Otherwise, it is auctioned to the highest bidder among those who have not yet been served cake, and the money is put in a pile. And so on, until everybody has been served. After that, the pile of money is split evenly among the participants.

Note that it is assumed here that cake cutting is difficult, but splitting an amount of money is easy. What do you guys think: is this a fair and envy-free procedure?

*Logician:* We should be able to tackle this with Parikh's logic, I suppose. But before we do that, it might be wise to have a look at the vast literature on this matter [4; 3; 2; 18; 21].

*Philosopher:* Yes, and let's not forget that rational action and the investigation of rationality is a classical theme in philosophy. Let me tell you a wonderful Indian story about the Mughal emperor Akbar and his minister Birbal [19] about the way in which knowledge and incentives affect a social algorithm. Birbal had asserted to the emperor that all wise (or clever) people think alike.

*Logician:* And then the emperor challenged him, right?

*Philosopher:* Right, so he suggested the emperor to order all men in Agra, the capital, to come at night to the palace grounds, and pour one potful of milk in the pool there, which was covered by a white sheet. The punishment for not doing so was severe, so one by one, all the residents of Agra came at night and poured a potful in the pool. And when the sheet was removed in the morning, it turned out that the pool was entirely full of water.

*Logician:* Of course.

*Philosopher:* Yes, and Birbal could explain to the emperor how this had to come about. "Your majesty, each man thought that if he, and he alone, would pour water instead of milk, it would not make much difference, and no one would notice. So they all did just that, for all your subjects are rational. And that's why your pool is full of water."

*Computer Scientist:* How wonderful!

*Philosopher:* By the way, there also is a story where Birbal acts exactly like Solomon. In the Hindu version, Ramu and Shamu claimed ownership of the

same mango tree, and decided to ask Birbal to settle the dispute. Birbal's verdict: "Pick all the fruits on the tree and divide them equally. Then cut down the tree and divide the wood." Ramu thought this was fair but Shamu was horrified, and Birbal declared Shamu the true owner.

*Computer Scientist:* It may interest you that Birbal's milk-pouring experiment was repeated by the psychologist Dan Batson, and with the same outcome. What Batson and his co-workers did [1] was set up a Birbal-like situation, where the subject was asked to flip a coin in private. The outcome of the coin toss was supposed to decide whether she herself or her team-member was scheduled for some unpleasant task. In collecting the results it turned out that these contradicted the laws of probability: more than 90 per cent of the subjects allotted the unpleasant task to their team member.

*Philosopher:* Why am I not surprised?

*Computer Scientist:* But the interesting thing was that all these cheating subjects duly reported that they had reached their decision in a fair way. Batson then tried to find out what incentive was needed to force the subjects to behave more honestly. It turned out that giving firm instructions about fairness and next placing them in front of a mirror was the only way to enforce ethical behaviour. Mind you, the subjects were psychology students, no doubt familiar with the one way mirror.

*Logician:* So what Batson was studying was not rational behaviour but the phenomenon of moral hypocrisy: our common tendency to believe ourselves to be more ethical than we truly are.

*Computer Scientist:* What still puzzles me about the Akbar and Birbal story is this: why did each of the cheating water pourers believe that he was the only cheater?

*Logician:* Well, they did as we all do, I suppose. They knew it didn't matter as long as they were not found out, so they gave it no further thought.

*Computer Scientist:* In any case, the story illustrates that reflection on social algorithms has a long history.

*Philosopher:* There is no doubt that the Akbar and Birbal stories go back a long time: emperor Akbar the Great ruled the Mughal Empire in the second half of the sixteenth century.

*Computer Scientist:* We talked briefly about auctions in connection with the

Solomon verdict. The study of auctions and their properties is part of an discipline called *mechanism design*. Surely, this also belongs to social software. You can find an overview in economics textbooks. See, e.g., Chapter 23 of [12]. Mechanism design deals with the problem of aligning agents's preferences so that the decision taken by the central authority is beneficial for the society. The best known example of a mechanism is that of a Vickrey auction, according to which the winner in a sealed-bid auction has to pay a price equal to the second highest bid.

*Logician:* Yes, we talked about that before.

*Computer Scientist:* Another area in social software where there is already a long and established tradition is voting theory. The mathematical study of voting procedures was started by Condorcet in the eighteenth century [5], and the literature has grown ever since. We surely know a lot about the advantages and disadvantages of different voting schemes.

*Philosopher:* It is interesting to reflect upon what motivated Condorcet to study voting procedures in the first place. He was struck by the fact that majority voting does not always lead to results that represent what the voters truly wish. A dangerous concept, by the way, but we will let that pass for now. In one and the same election, it is possible that a majority prefers $A$ over $B$, another majority prefers $B$ over $C$, and a third majority prefers $C$ over $A$. Majority preference is not transitive, and this is a flaw. Therefore, Condorcet proposes to start from pairwise comparisons between all alternatives. The Condorcet winner is the choice that beats all alternatives in pairwise comparisons.

*Computer Scientist:* Condorcet proposed organizing elections like chess tournaments. Not a very practical way to elect the president of France or the United States, if you ask me. Also, it is unfortunate that a Condorcet winner need not exist.

*Philosopher:* Not very practical for large-scale elections, indeed. And you are right that there is not always a Condorcet winner. Condorcet was aware of these facts, of course. But it is getting a bit chilly. May I propose we go inside and try to get some work done? Tomorrow, or at some later time, we can continue our discussion. Maybe we should try to come up with areas of social software where our combined expertise might make a difference.

*Computer Scientist and Logician:* Good idea. Let's think about it, and con-

tinue some other time.

## References

[1] C.D. Batson, E.R. Thompson, G. Seuferling, H. Whitney, and J.A. Strongman. Moral hypocrisy: appearing moral to oneself without being so. *Journal of Personality and Social Psychology*, 77(3):525–537, 1999.

[2] S. J. Brams and A. D. Taylor. *The Win-Win Solution.* W. W. Norton, New York, 1999.

[3] S.J. Brams and A.D. Taylor. *Fair Division: From Cake-Cutting to Dispute-Resolution.* Cambridge University Press, 1996.

[4] Steven Brams. Fair division. In Barry R. Weingast and Donald Wittman, editors, *Oxford Handbook of Political Economy.* Oxford University Press, 2005.

[5] Nicolas de Condorcet. *Essai sur l'application de l'analyse à la probabilité des décisions rendues à la pluralité des voix.* Imprimerie Royale, Paris, 1785.

[6] Karen S. Cook. *Trust in Society.* Russell Sage Foundation Publications, 2003.

[7] A.M. Duinhoven and K. Eykman. *Karel ende Elegast.* Nederlandse Klassieken Reeks. Prometheus and Bert Bakker, Amsterdam, 1997. Translation in modern Dutch. The 1486-1488 version in Middle Dutch can be found on `http://www.dbnl.org/tekst/_kar001kare01_01/`.

[8] Norbert Elias. *The Civilizing Process, Vol.I. The History of Manners.* Blackwell, Oxford, 1969.

[9] Norbert Elias. *The Civilizing Process, Vol.II. State Formation and Civilization.* Blackwell, Oxford, 1982.

[10] Francis Fukuyama. *Trust: The Social Virtues and The Creation of Prosperity.* Free Press, 1996.

[11] Martin L. Jones. A note on a cake cutting algorithm of Banach and Knaster. *The American Mathematical Monthly*, 104(4):353–355, April 1997. doi:10.2307/2974584.

[12] A. Mas-Collel, M. Whinston, and J. Green. *Microeconomic Theory.* Oxford University Press, 1995.

[13] J. Moore. Implementation, contracts, and renegotiation in environments with complete information. In J.-J. Laffont, editor, *Advances in Economic Theory — 6th World Congress*, volume I, Cambridge, 1992. Cambridge University Press.

[14] Eric Pacuit. *Topics in Social Software: Information in Strategic Situations*. PhD thesis, City University of New York, 2005.

[15] R. Parikh. Social software. *Synthese*, 132:187–211, 2002.

[16] Marc Pauly. *Logic for Social Software*. PhD thesis, ILLC, Amsterdam, 2001.

[17] Marc Pauly. Changing the rules of play. *Topoi*, 24:209–220, 2005.

[18] Jack Robertson and William Webb. *Cake-Cutting Algorithms: Be Fair If You Can*. A.K. Peters, 1998.

[19] Amita Sarin. *Akbar and Birbal*. Penguin India, 2005.

[20] Adam B. Seligman. *The Problem of Trust*. Princeton University Press, 2000.

[21] Jiri Sgall and Gerhard J. Woeginger. An approximation scheme for cake division with a linear number of cuts. *Combinatorica*, 27(2):205–211, 2007.

[22] Brian Z. Tamanaha. *On the Rule of Law*. Cambridge University Press, 2004.