THE ARCHITECTURE OF EVENT MEMORY

JOHN MORTON

Institute of Cognitive Neuroscience, University College, London

j.morton@ucl.ac.uk
Tel. (+44) (0)171 391 1156
Fax. (+44) (0)171 813 2835
17 Queen Square
London WC1N 3AR
United Kingdom

In this paper I am going to present some evidence concerning the nature of the processing architecture associated with event memory. I take it that the function of event memory is to allow us to learn from experience. This requires it to perform at least the following functions:

- to enable us to interpret what is going on around us
- to help plan actions and guide our behaviour

The starting point is the architecture shown in figure 1,which is conceived of within the framework of a particular model of event memory, Headed Records (Abeles & Morton, 1999; Kopelman & Morton, in press; Morton, 1990, 1991; Morton & Bekerian, 1986; Morton, Hammersley & Bekerian, 1985; Newcombe & Siegal, 1996; Wilkinson, 1988a, b).

Figure 1 is to be read in the following way. Information enters the *Input Buffer* from the environment. Some preliminary processing would have taken place by this stage. This information then passes on to the *Interpreter Buffer* where it can be processed further. The *interpreter* is to be seen as the collection of functions which operate on the information in the interpreter buffer. The purpose of this processing is to extract meaning, in order to understand what is happening in the environment. In pursuit of this overarching goal, records may be retrieved from the *Headed Records* system (event memory) into the Interpreter Buffer. A good candidate for this would be the record corresponding to the last time an event similar to the current event took place. The retrieval of a record would also follow the receipt of an explicit question, the answer to which could not be found in either of the buffers. Thus, if you were asked, “What did you have for dinner last night?”, you would retrieve the record corresponding to that event. Further details about the retrieval process may be found in the references above.

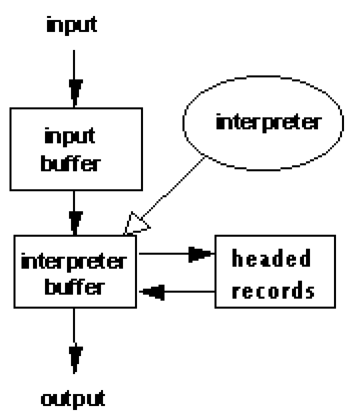

Figure 1 - A schematic of the structures involved in the creation and retrieval of event records

The next thing to consider is the formation of the records. These will be created from the contents of the buffers. Since Records are supposed to correspond to events, there needs to be an *event parser* function within the interpreter. This will decide how the stream of processed material shall be divided. We can imagine that, at suitable points, the parser triggers a record, at which time the contents of the Interpreter Buffer is dumped into the next slot in the Headed Records system. There is some evidence that such a step is triggered by change in scene (Morton, 1990; Wilkinson, 1988b). It would also make sense if records were triggered by change in protagonist or by a change in goals since the functions of interpreting what is going on around us and helping to plan actions would be affected by such factors.

The experiments I am going to present in the course of exploring the model have all been carried out on 3- and 4-year-old children. If one believes (as we do) that there is little or no development of memory structures with age, but that what develops are meta-abilities, then the use of pre-schoolers is a very effective way of having a clean look at structure without interference from the subject's strategies. Young children will be less likely to reflect on what is happening, would be more likely to pay full attention to the tasks we are presenting and would be much less likely to use complex meta-memoric strategies during the tasks which might circumvent our intent.

1 Interactions among the components - the Truck experiment

From the above description, it is clear that the Interpreter Buffer has a number of functions. It is used for cognitive operations on input, and for cognitive operations on event memory (Records). One might say that it is equivalent to (one of the functions of) Working Memory. As such, one might expect that the buffer has limited capacity, particularly for children. The truck experiment (Barreau, 1997) was designed to explore the interactions among the various components of the model.

The scenario consisted of two events. It involved two toy trucks - Truck A and Truck B. Truck A was in one location, and the children were invited to "drive" it into the next room, where they met a second experimenter. This sequence, we predicted, would constitute the first event, signalled by a change both in location and protagonist. The result of event (or sub-event) completion would be the creation of a record. In the second location, the children unloaded Truck A and then unloaded Truck B. Finally, both trucks were reloaded with different goods (*cupboards*, *people*, *food* and *boxes* were used) after which the children were asked to drive the Truck B back to the second room. Before this task was completed, they met the first experimenter again, who asked them the following three questions:

1. "The truck that you *carried*, what *was* in it?" - this referred to Truck A's past state, which will call *Apast*.

2. "The truck that you are *carrying*, what *was* in it?" - *Bpast*.

3. "The truck that you *carried*, what *is* in it now?" - *Apres*.

The questions were asked in two different orders, with 16 children in each group. The results are shown in Table 1. It will be seen that the two orders do not replicate each other, with Bpast not being available in the first order and Apres being missing from the second. Any attempt to account for such data will be constrained by two theoretical requirements:

- The Crucial items (Bpast and Apres) are still being actively processed *(Because they can be overwritten)*
- These items must be in different stores *(To allow selective overwriting of each individual item independently from the other)*

	Apast	Bpast	Apres
Order 1	12	1	12
Order 2	11	11	4

Table 1 - The results of the first Truck experiment (n=16 for each group)

The two orders in which the questions were asked were:
Order 1: (1) Apast? (2) Bpast? (3) Apres?
Order 2: (1) Bpast? (2) Apast? (3) Apres?

According to our initial assumptions, Apast would be in a record. As such it could not be overwritten, any error being in the process of retrieval. Beyond that, the simplest account we could come up with was that **Bpast** was in the Interpreter Buffer while **Apres** was still in the Input Buffer. With the further assumption that the Interpreter Buffer has a limited capacity, the account of the existing data is given in tables 2 and 3.

Material required for question answering has to be retrieved into the Interpreter Buffer. With Order 1, the first question relates to Apast which has to be retrieved from the record system into the Interpreter Buffer. Suppose we assume that the effect of that is to overwrite Bpast from the Interpreter Buffer. This corresponds to time 2 in table 2. We also assume that a record is made of the question answering (not shown in the diagram), at which time the Interpreter Buffer is cleared. When the question in relation to Bpast follows, there will be no relevant information in the system (time 3). However, information will be brought down from the Input Buffer for the Apres question.

time	Question Record	Buffer	Input Buffer	Interpreter System
1		Apres	Bpast	Apast
2	Apast?	Apres	Apast	Apast
3	Bpast?	Apres	-	Apast
4	Apres?	-	Apres	Apast

Table 2 - Flow of information for Order 1

For Order 2, Bpast is first queried. When that information is cleared from the Interpreter Buffer, Apres will be moved down from the Input Buffer (time 3 in table 3). When Apast is then questioned, it will overwrite Apres from the Interpreter Buffer. In this way, the Apres question will not find an answer in Order 2. Given that we can account for the data in Table 1 with a few assumptions, we can see what predictions can be made.

time	Question Record	Buffer	Input Buffer	Interpreter System
1		Apres	Bpast	Apast
2	Bpast?	Apres	Bpast	Apast
3		-	Apres	Apast
4	Apast?	-	Apast	Apast
5	Apres?	-	-	Apast

Table 3 - Flow of information for Order 2

The starting state of the system when the questions are asked is shown in figure 2 . Using this figure, we can first ask what would happen if Apres was questioned first. On the basis of the assumptions already made, it should be clear that retrieving Apres would have the effect of eliminating Bpast from the Interpreter Buffer. After answering the Apres question, then, children should not be able to answer the Bpast question. The Apast question should be unaffected since that information would be retrieved from the record.

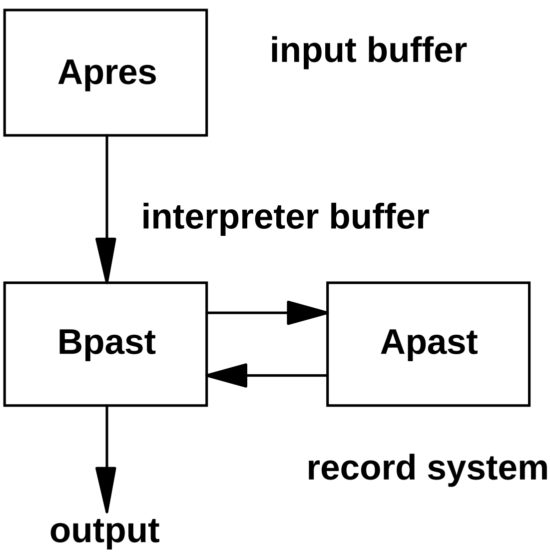

Figure 2 - The starting state of the system

The second test of the model is to consider how we could arrange for all three questions to be answered correctly. It is important to be able to do this in order to eliminate the alternative hypothesis that the children can only hold on the two of the answers. Within the model, the answer is clear. The first question should be about **Bpast**. This will lead to the Interpreter Buffer being cleared and **Apres** moving down into the Interpreter Buffer from the Input Buffer. Apres can then be probed and, finally, the **Apast** question can be asked.

In table 4 is the data arising from a repeat of the Truck experiment, using the two question orders:

Order 3: A pres? A past? B past?

Order 4: B past? A pres? A past?

	Apast	Bpast	Apres
Order 3	**11**	**0**	**9**
Order 4	**13**	**11**	**13**

Table 4 - tests of the model in figure 2 (n=16)

It will be seen that the data confirm the predictions made. We can tentatively conclude that we need a dual buffer system of the kind outlined, and that the Interpreter Buffer is of limited capacity for the 3- and 4-year-olds used in these experiments.

2 The Smarties Puzzle

We will now use the architecture to examine one of the most intriguing phenomena in cognitive development. In the standard Smarties experiment, the experimenter shows a 3-year-old child a Smarties tube and asks "What is in here?". The child answers "Smarties." The experimenter takes the cap off the tube and shows the child that the tube actually contains pencils. The top is replaced and after the child has confirmed the real contents they are asked what they had thought when they first saw the tube. A large proportion of 3-year-olds answer "Pencils" - the reality error. This error is much reduced with 4-year-olds and virtually vanished with 5-year-olds.

The usual explanation of this finding is that 3-year-olds have a problem computing with False Beliefs (Leslie, 1987; Perner, Leekam, & Wimmer,1987). This is almost certainly the case, but equally curious is that the child cannot recall what was in their mind 5 seconds earlier. In simulating this problem, we (Barreau, 1997; Barreau & Morton, 1999; Beaman & Morton, 1998) considered how the representation of the current state of the environment would get updated. Thus, when the child saw the Smarties tube she would set up a representation of the form in(tube,Smarties). In terms of the model in Figure 1, this representation would be in the Input Buffer. When the child was asked about the contents of the tube, this representation would be retrieved into the Interpreter Buffer and the answer given. In the previous section we supposed that when a question has been answered, the Interpreter Buffer would be cleared. The code in(tube,Smarties) would be retained in the child's representation of (what she believes to be) the current environment. When the child sees that the contents of the tube is actually pencils, this representation (since it is of the current environment) will have to be updated to in(tube,pencils). The former code, being incorrect, would have to be deleted, since it does not apply any more. When the child is then asked what they had thought was in the tube when they first saw it, there are three possible sources of information. The first, the various buffers, are ruled out because, as described, the relevant information has been deleted. The second way is to look at the environment, see the tube and work out that would have thought that there had been Smarties in it since it is a Smarties tube. However, this approach involves the child computing with a false belief, which we have other reasons for supposing is beyond them. The final possibility is that they can retrieve the record we suppose them to have created when they answered the initial question. Since they mainly fail to come up with the correct answer we can assume that the record, if it exists, cannot be addressed with the question "What did you think ...?"[1]

Within the model, how can we overcome these problems? One way is to induce the system into creating a record of the initial state (or, rather, the child's false belief, in(tube, Smarties), as to its nature) in a way which is retrievable. We did this with what we call The Bag Experiment. The procedure begins in the usual way: the child is shown the Smarties tube and asked what she thinks is in it. Then a bag is produced. The Smarties tube is emptied into the bag without the child being able to see the contents. The child is then asked what she thinks is in the bag. All children say "Smarties". The bag is then opened and the pencils shown to the child. The pencils are put away and the child is asked: "Before I opened the bag what did you think was in it?" Finally the tube is produced and the child is asked a variety of questions including what they had thought was in the tube when they first saw it.

---

1 A more recent study has shown that when children are asked "What did you say ..?", they are twice as likely to give the correct answer (Morton & Dillon, in preparation).

This sequence of events is cryptically referred to in table 5 where the fates of the various representations are traced. Thus, the input buffer starts with **in(tube,Smarties)** - written as in(t,S) in table 5. This representation gets updated to **in(bag,Smarties)** at the time of the transfer of contents to the bag (time 3). At this time, we suppose that the event parser triggers the creation of a record containing, among other things, **in(tube,Smarties)**. When the pencils are discovered, the representation of the current state will be changed to **in(bag,pencils)**, with **in(bag,Smarties)** being destroyed. When the question about the bag is then asked, there will be no information readily available concerning Smarties (time 6), and so the false belief error will be made, as in the original Smarties experiment. When the tube is produced, however, we hope that the child will be able to retrieve the record of the earlier state, and discover reference to Smarties. The data are shown in table 6. These are the summed data from experiments 1 and 2 in Barreau and Morton (1999). About a third of the children were correct in answer to the bag question, about the same proportion traditionally found in the Smarties experiment. When the same children were asked what they had first thought the contents of the tube to be, however, nearly 75% more of them gave the correct answer.

Time	**Event Buffer**	**Input Buffer**	**Int'p'ter System**
1 tube	in(t,S)		
2 "in tube?"	in(t,S)	in(t,S)	
3 bag	in(b,S)	-	in(t,S)
4 "in bag?"	in(b,S)	in(b,S)	in(t,S)
5 pencils	in(b,p)		in(t,S)
6 "in bag?"	in(b,p)	in(b,p)	in(t,S)

Table 5 - Sequence of events and supposed flow of information in the bag experiment

We can conclude from this that we were correct in supposing that the bag transfer would trigger the formation of a record which would be later accessible. It also turns out that false belief is not an overriding issue - if the correct information can be found without involving complex computation, the false belief can be reconstructed. All of this reinforces our hypothesis that the current state of certain aspects of the environment is registered in a buffer in such a way that when the environment changes, this representation is destructively updated.

3 Current State Buffer - Tracking Teddy

What we have been talking about is a function concerning the current state of the environment. This will have to track key elements of the environment, which can be observed or inferred, and there will be destructive updating for locations, contents etc. We assume that these functions are automatic. In the previous discussion we assigned this function, which we will call the Current State Buffer, to the Input Buffer (as defined in the Truck experiment). In the final section we will examine this structural assumption as well as establishing the independence of the Current State Buffer from Working Memory or the Interpreter Buffer.

The experimental design we used to study the Current State Buffer had two main components (Abeles, 1999; Abeles & Morton, in press). The basic task involved placing a number of objects, such as a crayon, a toy cat and a piece of Lego, into separate containers. The containers included a bowl, a box and a cup. The child is asked to observe this and to remember where everything is hidden. This is a prototypical Working Memory task (Baddeley, & Hitch, 1974; Baddeley, & Lieberman, 1980; Hitch, Halliday, Schaafstal, & Schraagen, 1988). The second component of the task was Teddy, a very small stuffed toy which was animated for one group of subjects and not for the other. For the former group, we would expect Teddy to be tracked by the Current State Buffer. This condition is called the Character condition. It contrasts with the Object condition, in which Teddy is treated in the same way as the other objects.

The experiments were carried out with 3- and 4-year-olds. To begin with, we established that three pairings of object and receptacle were too much for the 3-year-olds and four pairings were too much for the 4-year-olds. Any further pairing which involved Working Memory would produce further interference. If, however, the extra pairing involved the Current State Buffer, and this was independent of Working Memory, there would be no further interference. This is essentially a dual task methodology: i.e. a Current State Buffer task concurrent with a Working Memory task. Performance on one task should not interfere with that on the other.

In the core part of the task, an Emu glove puppet tidies away toys at the front of the table into receptacles at the back of the table. With each toy, the child is reminded to note where the object is so that they can remember later. When all the objects had been hidden, the children were tested with probed recall, e.g. "where is the crayon?" In one experiment the children were allowed to talk and in another they pointed. This made no difference to the data.

In the Teddy as Character condition, Teddy interacted with the child and then went to sleep in one of the receptacles. The children were not asked to note where he had gone. That is, any recall of Teddy's location would be incidental. In the Teddy as Object condition, Teddy was just another object, tidied away into one of the receptacles. As in the Character condition, Teddy was always hidden first. The location of Teddy was probed last in both conditions. In some experiments there was another animate character, Simba, who interacted with the child and then went to sleep in one beneath the table. In other replications the Teddy as Object condition had no animate character. This made no difference to the data. The data we report here involve 58 subjects: 28 three year olds, 28 four year olds. As already mentioned, 3- year-olds were given 3 basic pairings and the 4-year-olds were given 4 pairings. There is an additional pairing of Teddy, either as Object or as Character. After Emu had tidied away the toys, the children were probed as to the location of the objects and of Teddy. The data for recall of the location of Teddy are given in Figure 3. The first striking feature of this data is the perfect recall of Teddy's location in the Character condition. This is even more remarkable considering firstly that the recall was incidental, and secondly that recall of Teddy followed recall of the objects. The difference between the two conditions is highly significant ($F(1,52)=17.67$, $p <0.0001$).

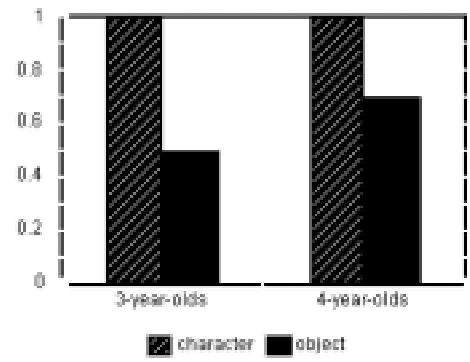

Figure 3 - probablity of recall of Teddy's location

Note that there are a variety of theories which would predict that Teddy's location would be better recalled in the Character Condition than in the Object Condition (see Abeles & Morton, in press). However, so far as we can see, all of them would also predict that the price for the superiority with Teddy would be a reduction in performance on the objects in the Character Condition. This is because of the limited capacity of Working Memory. The data for the objects is given in figure 4. It can be seen that, for both age groups, the locations of more objects are recalled in the Character Condition than in the Object Condition. This difference is highly significant ($F_{(1,52)} = 8.81$, $p <0.0045$). This combination of data, involving perfect incidental recall of Teddy's location in the Character Condition, plus better performance on objects when Teddy is Character than when he is Object would not be predicted by any other theory we are aware of. It leads us to conclude that the location of Teddy is in the Current State Buffer, whereas the location of objects is in Working Memory. We conclude that the Current State Buffer is independent of Working Memory, that it was responsible for tracking Teddy, and that it did this automatically. We assume that this function of the Current State Buffer only applies for significant characters, such as Teddy, or objects such as Smarties.

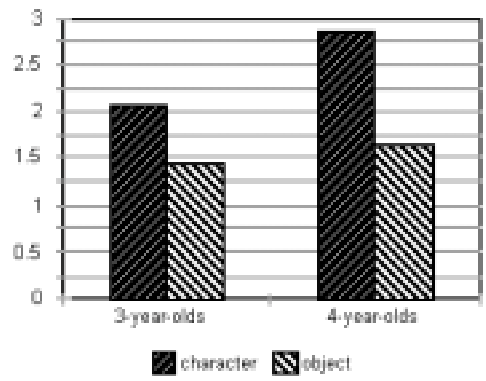

Figure 4 - Mean number of objects recalled correctly

It has been assumed up to now that the functions of the Current State Buffer are carried out in the Input Buffer of Figure 1. We can now examine this proposition. In figure 5 we diagram the two buffers with the supposed contents in the Tidy Emu paradigm. The Record system has been omitted as it is not relevant. In the tidy Emu paradigm, Teddy goes to sleep first. The objects are hidden afterwards. We can symbolise this as T123 - where 123 refers to the three objects used with the 3-year-old subjects. We can then ask the question as to why Teddy isn't affected at all by the need for the other information to come in. Remember that recall of Teddy is perfect in the Character Condition. There seem to be three alternatives:

1. the input buffer has unlimited capacity.

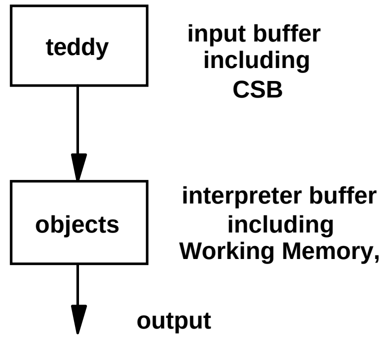

Figure 5 - Supposed storage of locations of Teddy and objects

2. the location of Teddy is interference resistant within the input buffer.
3. the Current State Buffer needs relocation.

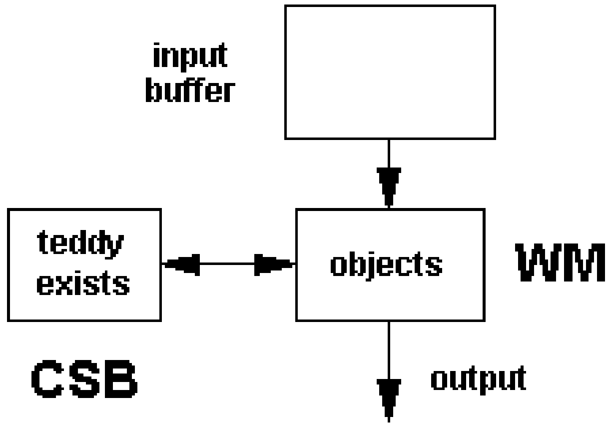

Figure 6 - Alternative architecture with the Current State Buffer (CSB) separate from the input buffer.

There are a number of arguments against the first two options in the spirit of our current framework (Abeles, 1999: Abeles & Morton, in preparation). In addition we should consider that information can enter the Current State Buffer by virtue of inference, as was indicated in the original Smarties experiment and the bag variant. In both cases, the representation **in(tube,Smarties)** is inferred rather than being the result of direct observation. An architecture such as that in Figure 6, then, would have something to recommend itself.

By analogy with the discussion of the Truck experiment, this suggestion leads to the prediction that if Teddy's location is retrieved first, then there should be loss of memory for the objects in the Teddy as Character condition through interference in Working Memory (here equated with the interpreter buffer). This experiment was carried out with a group of 3-year-olds. The comparison is with the average of three experiments where the location of Teddy was probed last - which we designate as a T123-123T design. In these experiments, an average of 1.95 objects were correctly located. In the new experiment, which we can designate as T123-T123, the average number of objects correctly located was 1.14, significantly worse. Probing Teddy first indeed seems to involve bringing information through Working Memory, Thus, far, the revised architecture is satisfactory.

Let us now see what would happen if Teddy's location is changed after the objects are hidden. With this architecture, there should be loss of information concerning the objects. The experimental design would be 123T-123T. Teddy would be introduced to the children first, in the usual way. Next, the objects would be hidden. Teddy then goes to sleep. This situation is shown in figure 7. For the change in state of Teddy to get to the Current State Buffer, it must pass through the already overloaded Working Memory. There should, then, be a reduction in the object score. A further 16 3-year-olds were run in this experiment and they managed to recall an average of 2.07 objects, which compares with the figure of 1.95 objects found previously. Teddy's location was, of course, recalled correctly by all children. Thus, the update of Teddy's location cannot pass through Working Memory.

We conclude that the Current State Buffer must be directly updated from the Input Buffer, and currently propose the architecture shown in figure 8. This incorporates a number of features derived from our experiments.

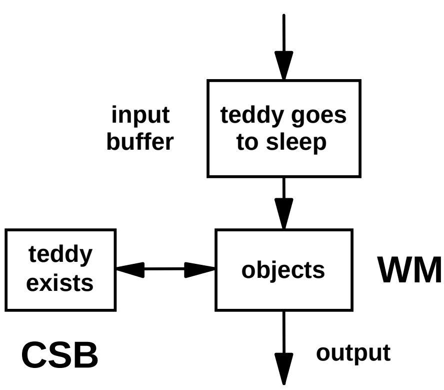

Figure 7 - What hapens with this architecture if the objects are hidden before Teddy?

1. Records are triggered by changes in significant features of the environment. We view the interpreter as having a parsing function which enables this. Event parsing can be triggered by changes in environment, protagonist or goal.

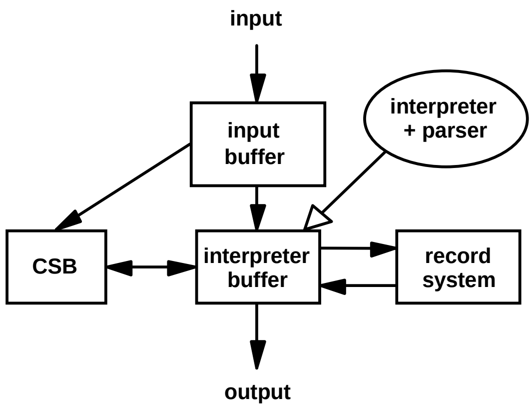

Figure 8 - The latest proposal as to the architecture involved in event memory

2. All buffers have limited capacity and contents can be interfered with[2].
3. The Current State Buffer is independent of the Interpreter Buffer (Working Memory).
4. The CSB is updated directly from the input buffer. We should note that the current architecture predicts that if the Current State Buffer is updated as a result of inference, then it should interfere with the contents of Working Memory. Productive predictions of this kind are essential if the model is to have value.

References

1. Abeles, P. (1999) The Development of the Current State Buffer. Unpublished Doctoral Thesis. University College London.
2. Abeles, P., & Morton, J. (1999). Avoiding misinformation: reinstating target modality. *Quarterly Journal of Experimental Psychology, 52A,* 581-592.
3. Abeles, P. & Morton, J (in press) Keeping Track: The Function of the Current State Buffer *Cognition.*
4. Baddeley, A.D. & Hitch, G.J. (1974). Working memory. In G. Bower (Ed.), *The Psychology of Learning and Motivation, vol. 8,* (pp. 47-90). New York: Academic Press.
5. Baddeley, A. D., & Lieberman, K. (1980). Spatial working memory. In R. S. Nickerson (Ed.), *Attention and Performance, 8,* (pp. 521-539). Hillsdale, NJ: Lawrence Erlbaum Associates.
6. Barreau, S. & Morton, J. (1999) Pulling Smarties out of a bag. *Cognition, 73,* 65-87
7. Beaman, C. P. & Morton, J. (1998) Modelling Memory - Updating characteristics of 3- and 4- Year Olds. *Proceedings of the Second European Conference on Cognitive Modelling, Nottingham.*
8. Hitch, G.J., Halliday, M.S., Schaafstal, A.M., & Schraagen, J.M.C. (1988). Visual working memory in young children. *Memory and Cognition, 16,* 120-132.
9. Kopelman, M & Morton, J. (in press) Psychogenic amnesia - functional memory loss. *In* G. Davies & T. Dalgleish, *Recovered Memories: Seeking the Middle Ground.* Chichester, Wiley.
10. Leslie, A.M. (1987). Pretence and representation: The origins of "theory of mind". *Psychological Review, 94,* 412-426.
11. Morton, J. (1990) The development of event memory. *The Psychologist,* 3, 3-10.
12. Morton, J. (1991) Cognitive pathologies of memory: a headed records analysis. In W.Kessen, A. Ortony & F. Craik (eds) *Memories, Thoughts and Emotions: Essays in Honor of George Mandler*. New Jersey: Erlbaum, 199-210.
13. Morton, J. (1997) Free Associations with EPS and memory. *Quarterly Journal of Experimental Psychology, 50A,* 924-941.
14. Morton, J. & Bekerian, D.A. (1986) Three ways of looking at memory. *Advances in Cognitive Sciences I*, Chichester: Ellis Horwood Ltd, 43-71.
15. Morton, J., Hammersley , R.H. & Bekerian, D. A. (1985). Headed records: A model for memory and its failures. *Cognition, 20,* 1-23.
16. Newcombe, P.A. & Siegal, M. (1996) Where to look first for suggestibility in young children. *Cognition*, **59**, 337-356.
17. Perner, J., Leekam, S. & Wimmer, H. (1987). Three year olds' difficulty with false belief: The case for a conceptual deficit. *British Journal of Developmental Psychology, 5,* 125-127.

[2] Abeles (1999) has shown that the Current State Buffer has limited capacity.

18. Wilkinson, J. (1998a) Context effects in children’s memory. In: Gruneberg, M.M., Morris, P.E., & Sykes, R.N. (eds) *Practical Aspects of Memory Current Research and Issues: Volume I*. Chichester: John Wiley & Sons. Pp. 107-111.
19. Wilkinson, J. (1998b) *Preschoolers Event Memory: Methods for Facilitating Recall*. PhD Thesis, University College, London