Causal Cognitive Architecture 1 (CCA1): Integration of Connectionist Elements into a Symbolic Framework

Howard Schneider
Sheppard Clinic North, Toronto, Canada

BICA*AI 2020
2020 Annual International Conference on Brain-Inspired Cognitive Architectures for Artificial Intelligence
Joint VR Track October 10-11, 2020

TEMPORARY NOTE -- REMOVE THIS SLIDE BEFORE PRESENTATION

DUE TO CHANGES IN CONFERENCE FORMAT IN 2020 THESE SLIDES TO BE PRESENTED VIA JOINT VIRTUAL REALITY TRACK OCT 10-11/2020

MATERIAL BELOW CAN BE PRESENTED IN SESSION 15 MINUTES – 60 MINUTES
MATERIAL PRESENTED WILL BE ADJUSTED TO TIME ALLOCATION

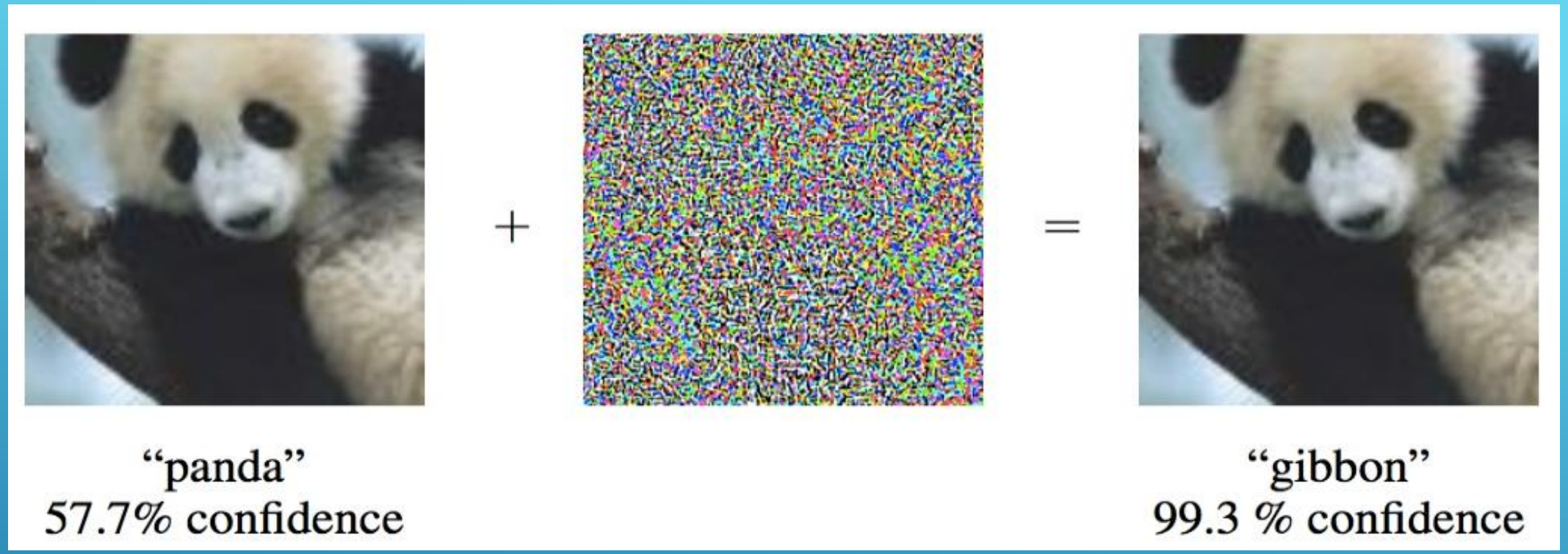
TEMPORARY NOTE -- REMOVE THIS SLIDE BEFORE PRESENTATION

Causal Cognitive Architecture 1 ("CCA1")

- Why do work on this subject? ←
- Why yet another cognitive architecture?
- What is the CCA1?

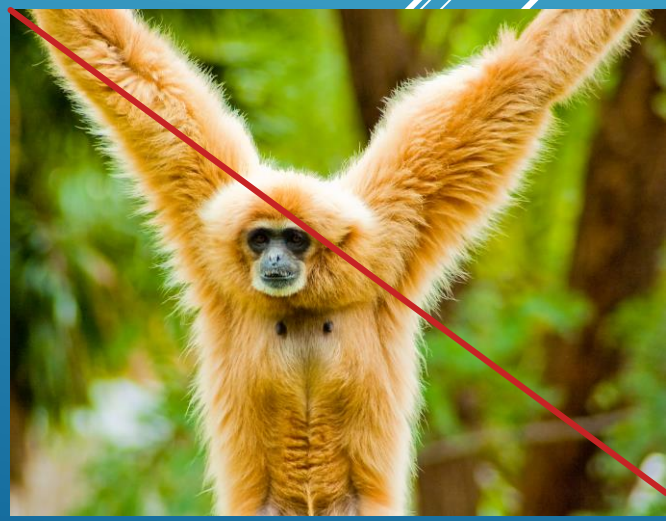
The problem: The *Neural Symbolic Gap*

- **Neural Network** – phenomenal image processing and reinforcement learning
- **Child** – phenomenal causal learning with few examples (eg, Gopnik)



Goodfellow,I.J., Shlens,J. and Szegedy,C. (Google Mountainview), Explaining and Harnessing Adversial Examples, ICLR 2015.

It’s still a Panda – and the 3 year old boy would know this!!
(and….. 3 year old only needs 1 or 2 photos for training, not 1000s)



Deep Learning Neural Network	3 Year Old Human Child
Pattern Recognition →Recognize the World	Model Building +also Pattern Recognition →Explain the World
Need 1000's examples for learning	A few examples enough

Causal Cognitive Architecture 1 ("CCA1")

- Why do work on this subject?
- **Why yet another cognitive architecture? ←**
- **What is the CCA1?**

1 slide primer: “What is a Cognitive Architecture?”

a theory about structure/functioning of <biological> mind

+

formalized enough to be basis of a computer program

Causal Cognitive Architecture 1 (CCA1)

- Mesoscopic brain inspired cognitive architecture – good balance of low/mid level and high level components and features
- One pragmatic solution to the neural-symbolic problem

Inspiration from other Cognitive Architectures

- ACT-R – central procedural module through limited capacity controls peripheral modules
- ART – attentional vs orienting system
- CLARION – connectionist + rule induction
- Recommendation Arch – limit information handling resources
- SOAR – procedural/reinforcement learning, semantic learning, episodic learning

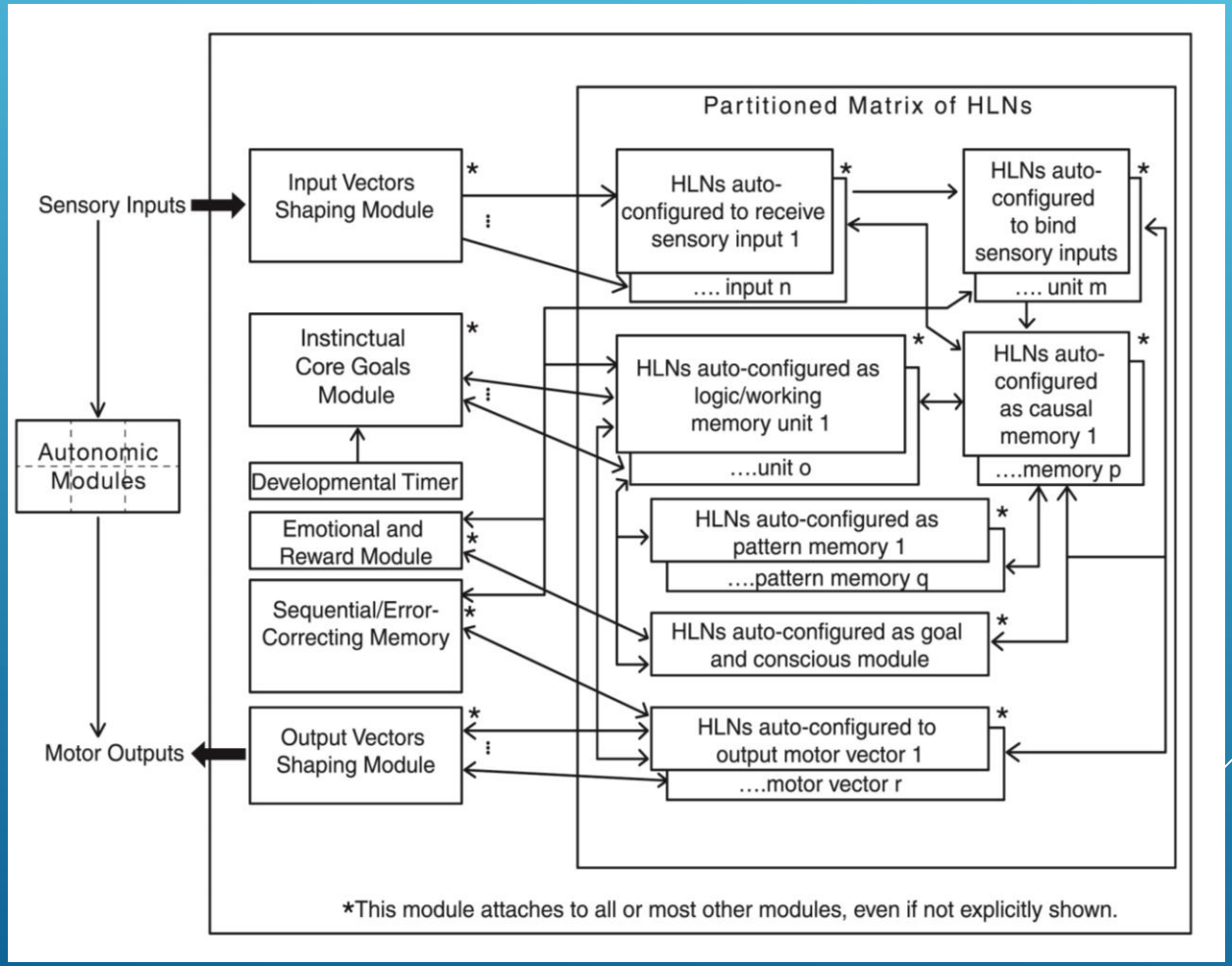
Why do this work? -- Hopes for the CCA1:

1. Technology producing functional behavior (subsymbolic operation)
2. Technology producing AGI (causal symbolic operation)
3. AGI: simple human to superintelligence levels
4. Useful insight into mesoscopic brain function and pathology

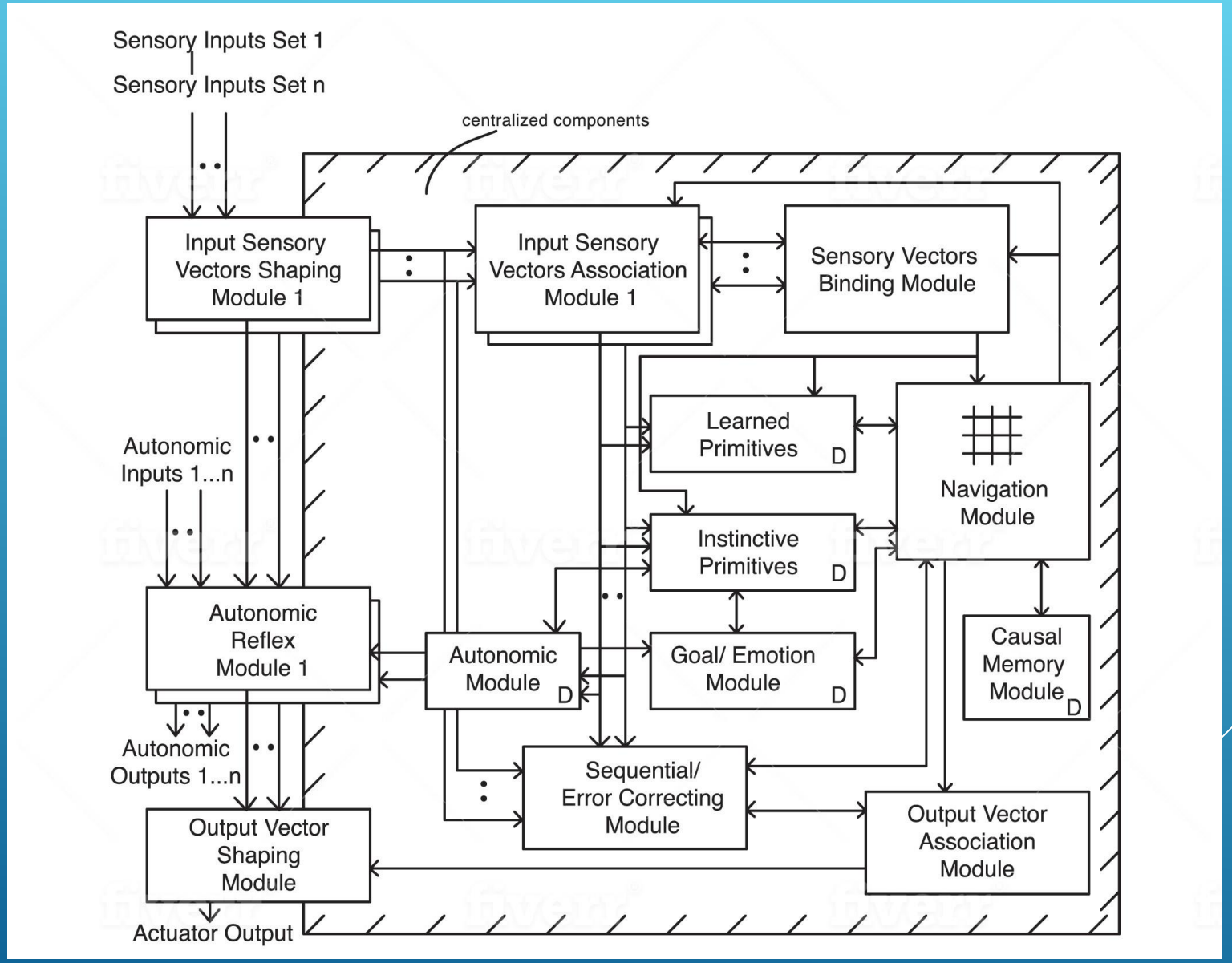
Causal Cognitive Architecture 1 ("CCA1")

- Why yet another cognitive architecture?
- Why do work on this subject?
- **What is the CCA1?** ←

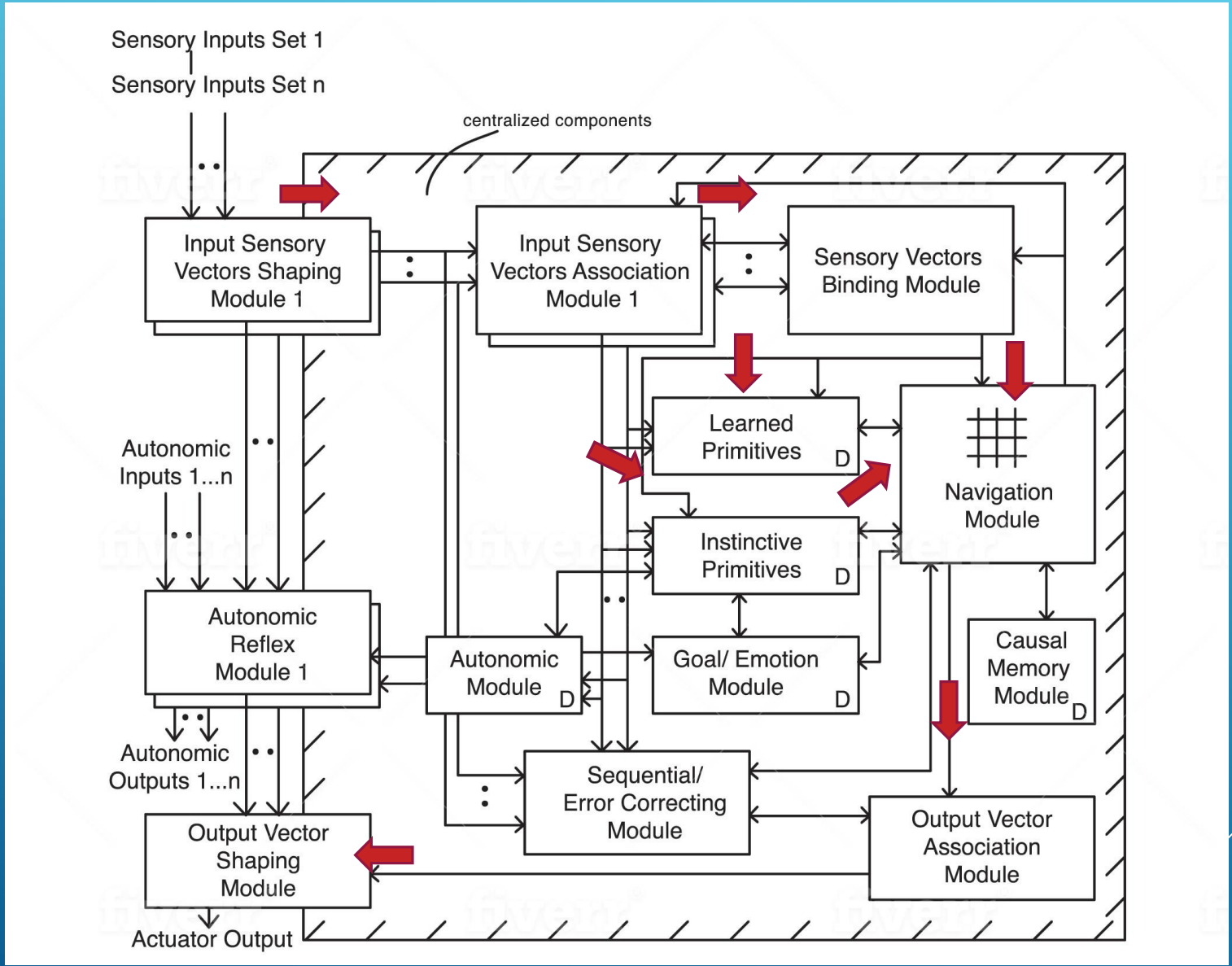
Derived from my previous work on the Meaningful-Based Cognitive Architecture



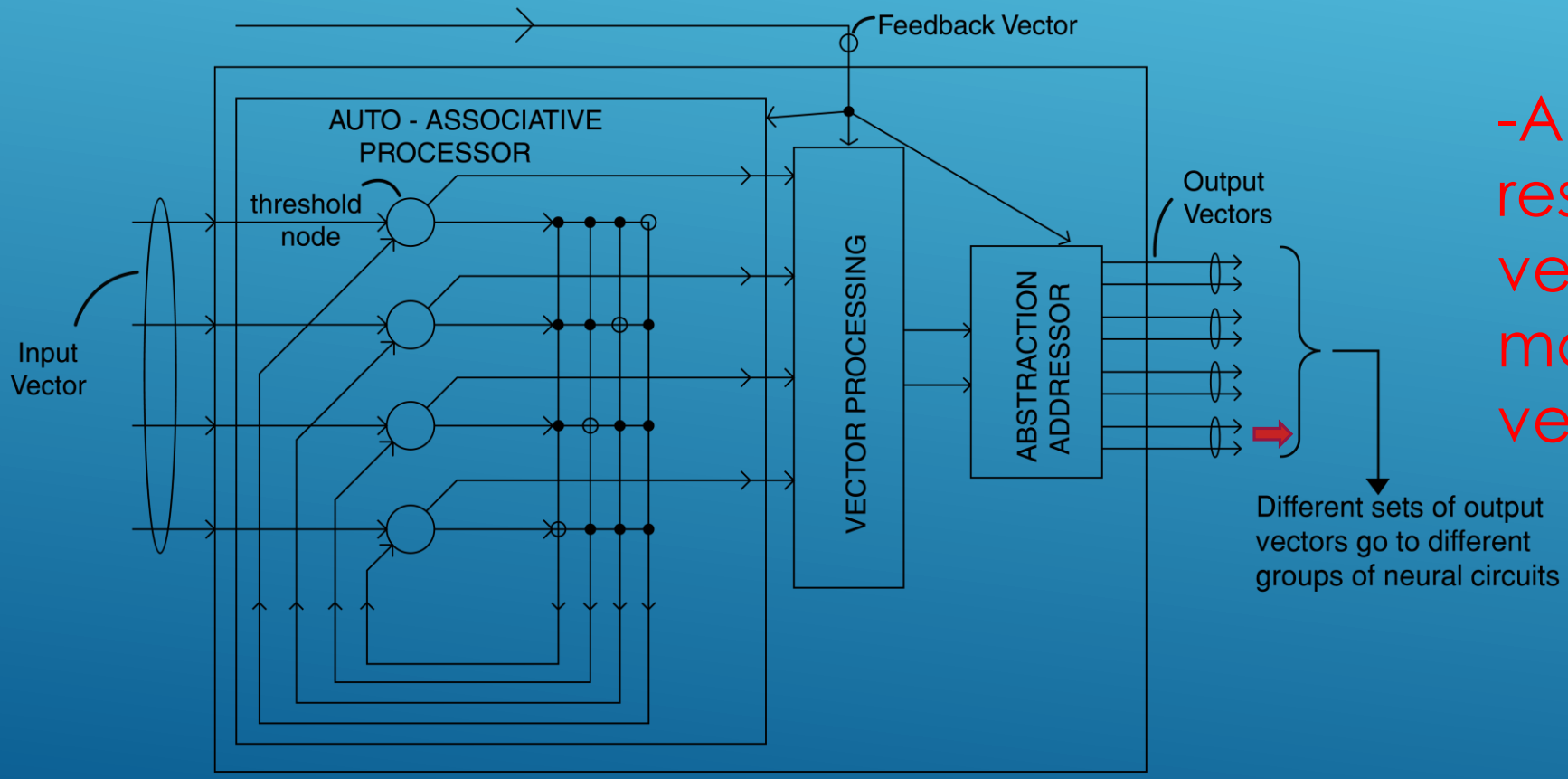
Causal Cognitive Architecture 1 (CCA1)



Sensory Inputs -> Association Modules & Binding ->Navigation Module -> Output Process -> Motor Outputs



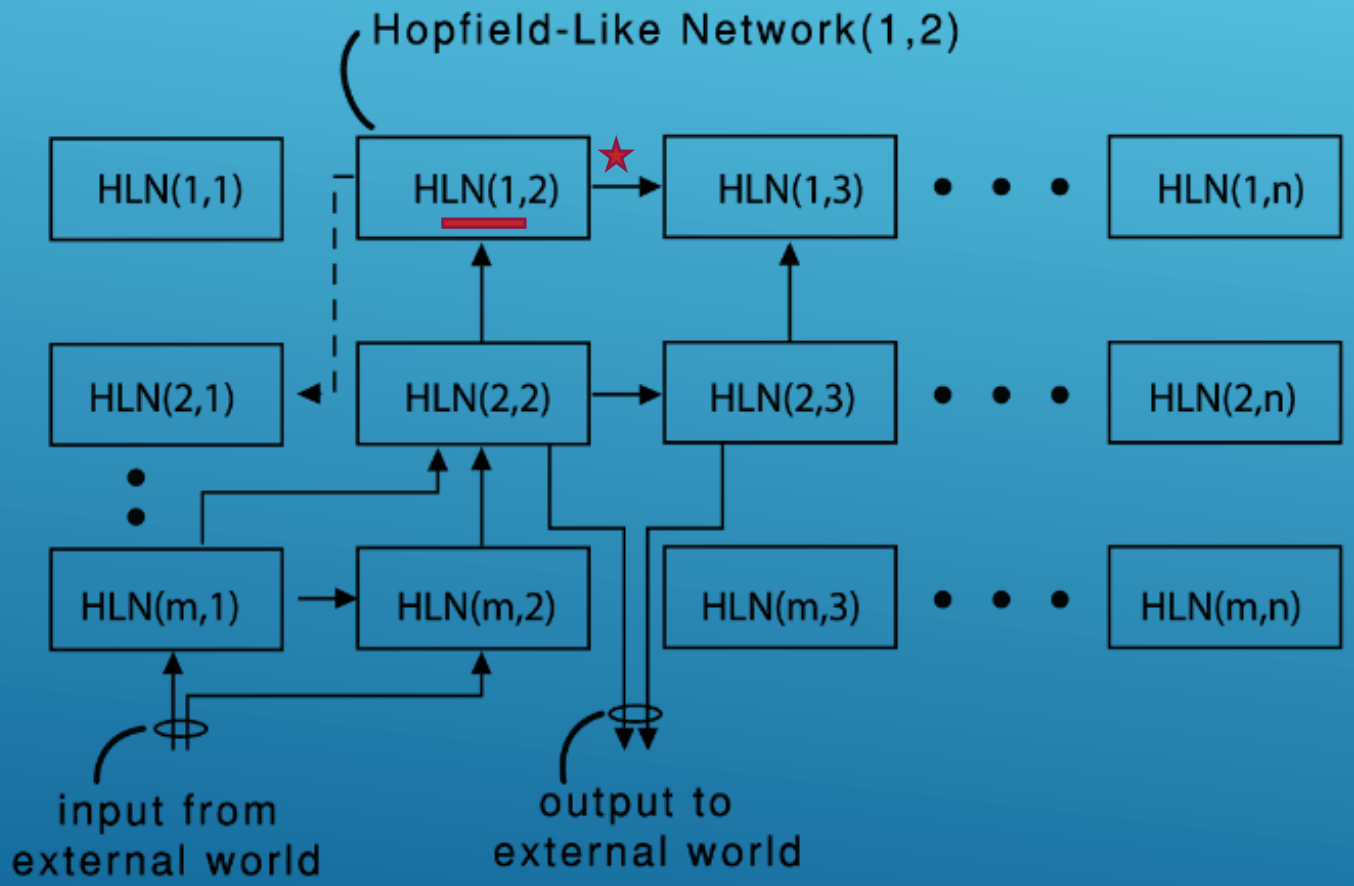
Basic Unit: **Hopfield-*like* Network** (‘**HLN**’)



-Auto-Associative Processor is a pattern recognizer

-Abstraction addressor in response to the feedback vector, decides which of many possible output vectors to use

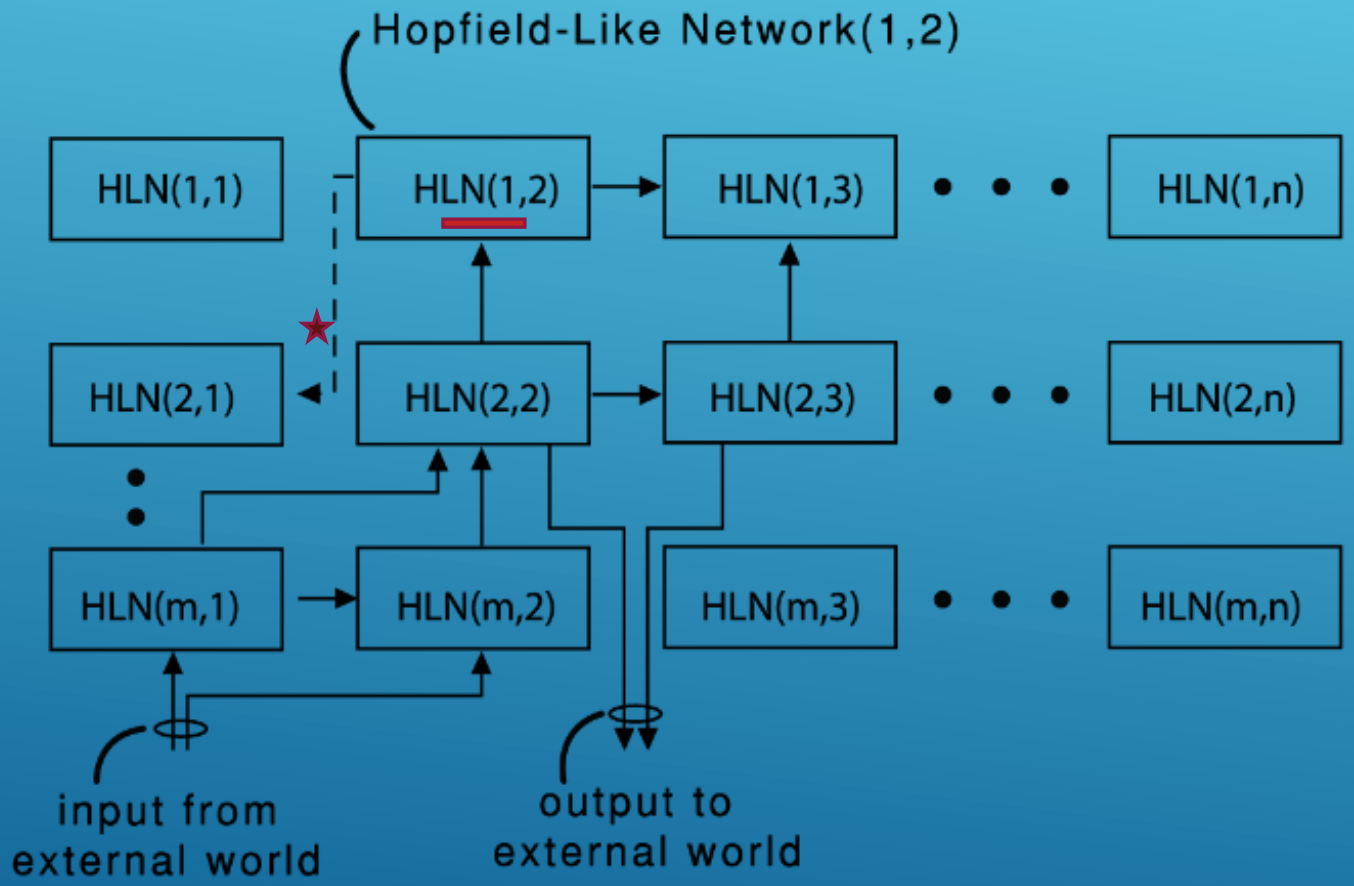
Weights between HLNs ← *extreme rapid reconfigurations*



← **HLN(1,2)** attaches to both **HLN(1,3)** and **HLN(2,1)**

Which HLNs do we connect to other HLNs?

Try to Maximize Meaningfulness



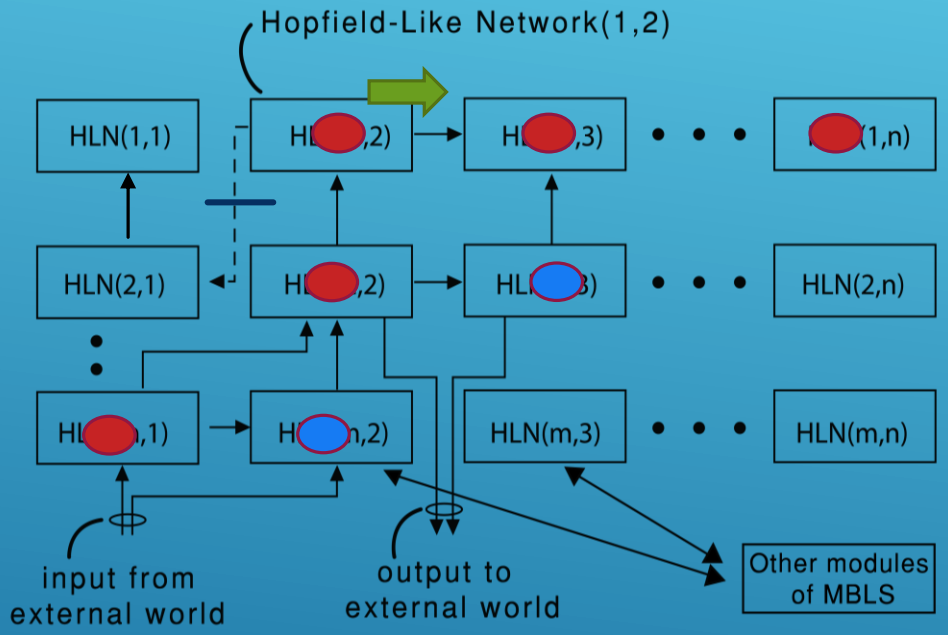
What is *Meaningfulness* ?

$H = -\sum_i P(x_i) \log_2 P(x_i)$ ←*Shannon entropy*

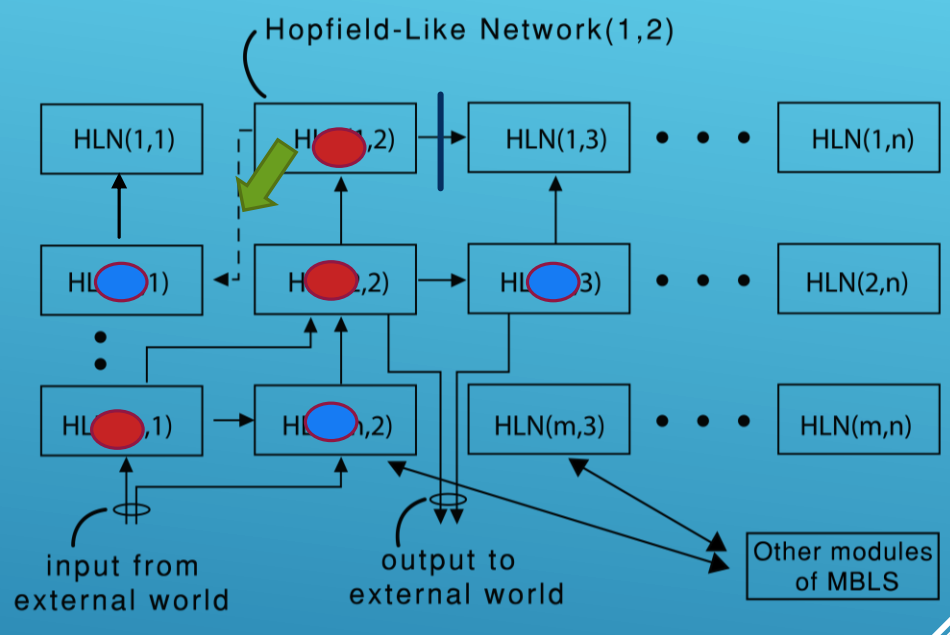
$M = 1/H$ ←*Meaningfulness*

Unfair coin: Tails, Tails, Heads, Tails, Tails, Tails
Therefore, low entropy, high meaningfulness

Meaningfulness – via Shannon Entropy



Reconfiguration A



Reconfiguration B

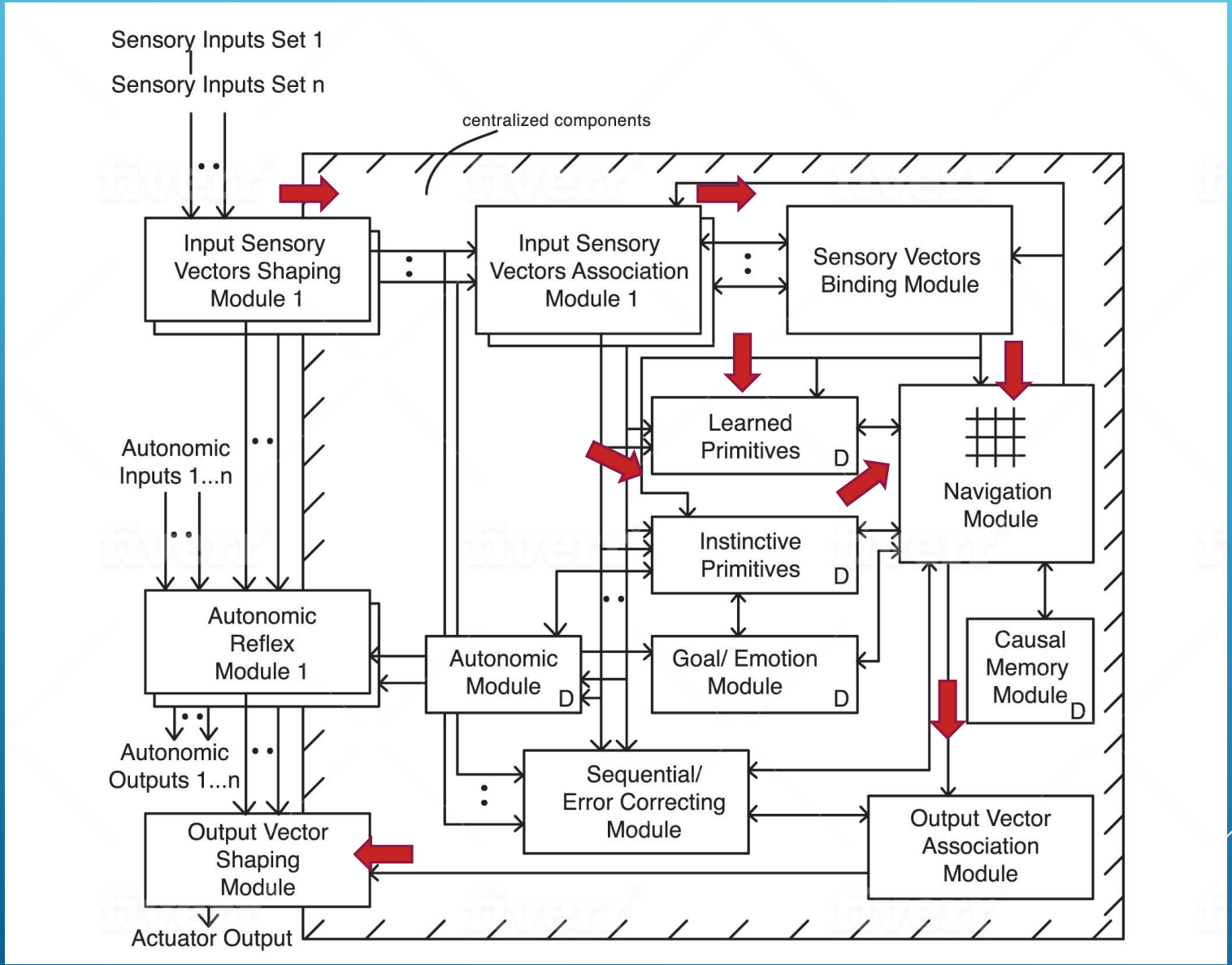
5 HLNs 'On', 2 HLNs 'Off' vs. 3 HLNs 'On', 3 HLNs 'Off'

p(ON)=5/7, p(OFF)=2/7 vs. p(ON)=3/6, p(OFF)=3/6

H= 0.86 → **M=1.2** vs. H=1.0 → **M=1.0**

(via Counting: 5 on vs. **via Counting: 3 on)**

CCA1 – Let's see some examples how it works…..



Hiker lost in the woods…..

Robot goes to the forest to save the hiker….

← Robot

← *Controlled by an CCA1*

As convenience, I will say: ***"CCA1"***

"CCA1" *= Robot + CCA1*

Choose pre-causal functioning of CCA1

Command Prompt - cca1_2020

```
Please choose type of "hippocampus"/"brain" which, of course,
 only loosely approximates the biological equivalent:
1. Lamprey hippocampal/brain analogue
2. Fish hippocampal/telencephalon analogue
3. Reptile hippocampal/pallium analogue
4. Mammalian hippocampus - note: meaningfulness, precausal
5. Human hippocampus - note: meaningfulness plus full causal features
6. Augmented Human level 1 - simultaneous multiple navigational threads
7. Augmented Human level 2 - algorithm center in each navigational module
Please make a selection:
```

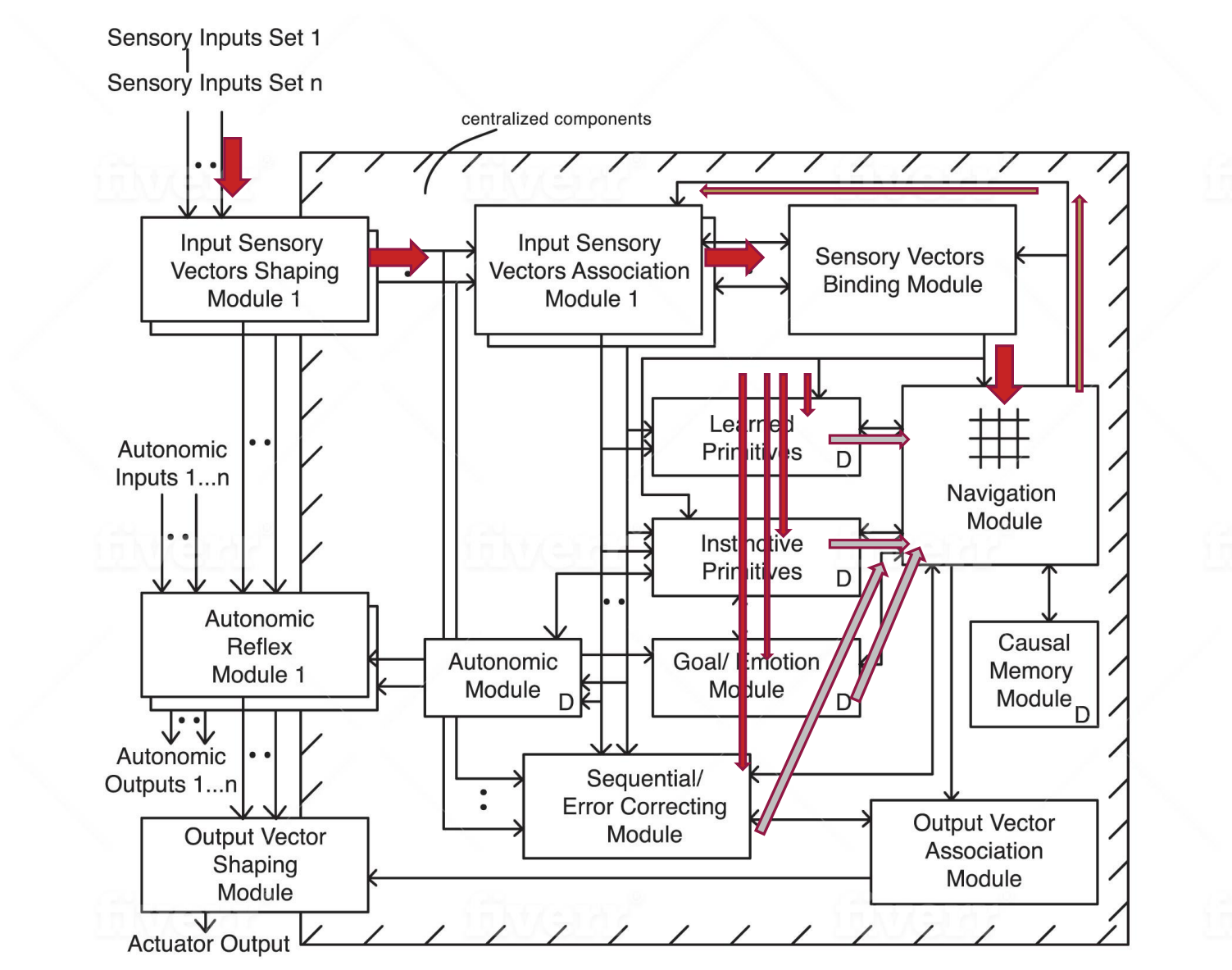
CCA1 must navigate to the lost hiker's square

Command Prompt - cca1_2020

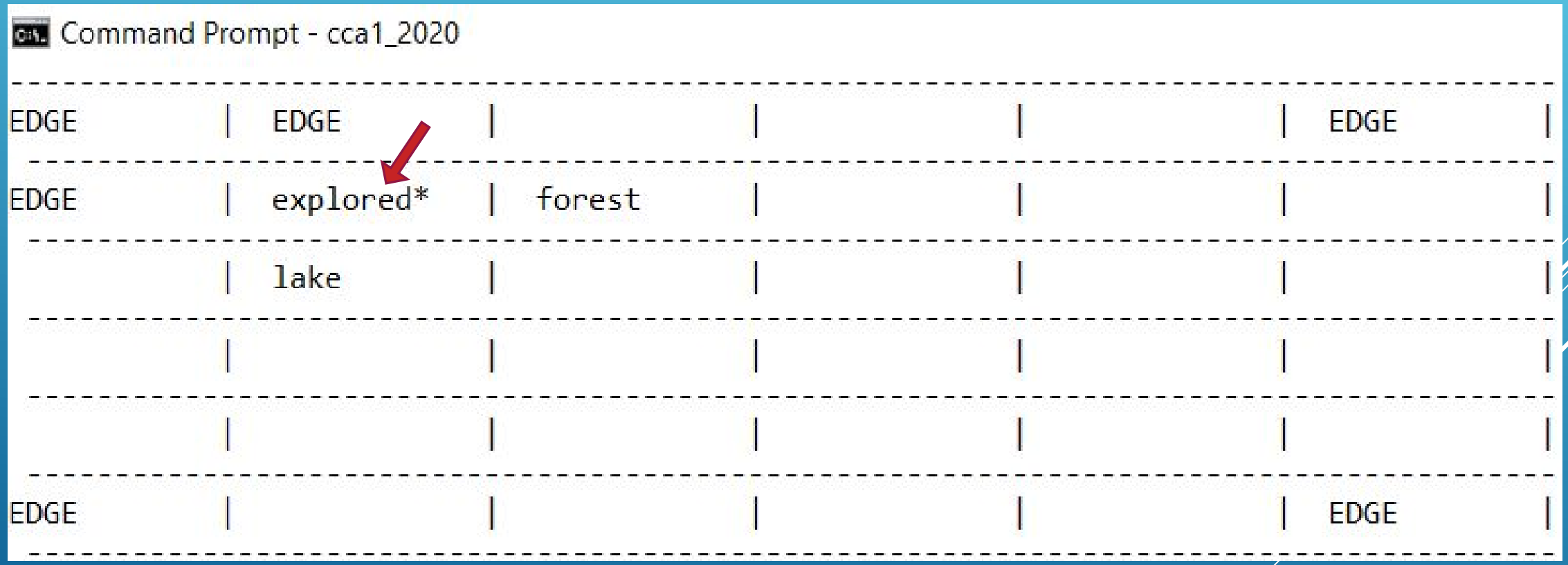
```
hiker position set to:  4 2

Bird's-Eye View of Forest (CCA1 does not have this view)
```

EDGE	EDGE	EDGE	EDGE	EDGE	EDGE
EDGE	CCA1 *	forest	sh_rvr	forest	EDGE
EDGE	lake	forest	forest	forest	EDGE
EDGE	forest	wtrfall	forest	forest	EDGE
EDGE	forest	hiker	forest	forest	EDGE
EDGE	EDGE	EDGE	EDGE	EDGE	EDGE

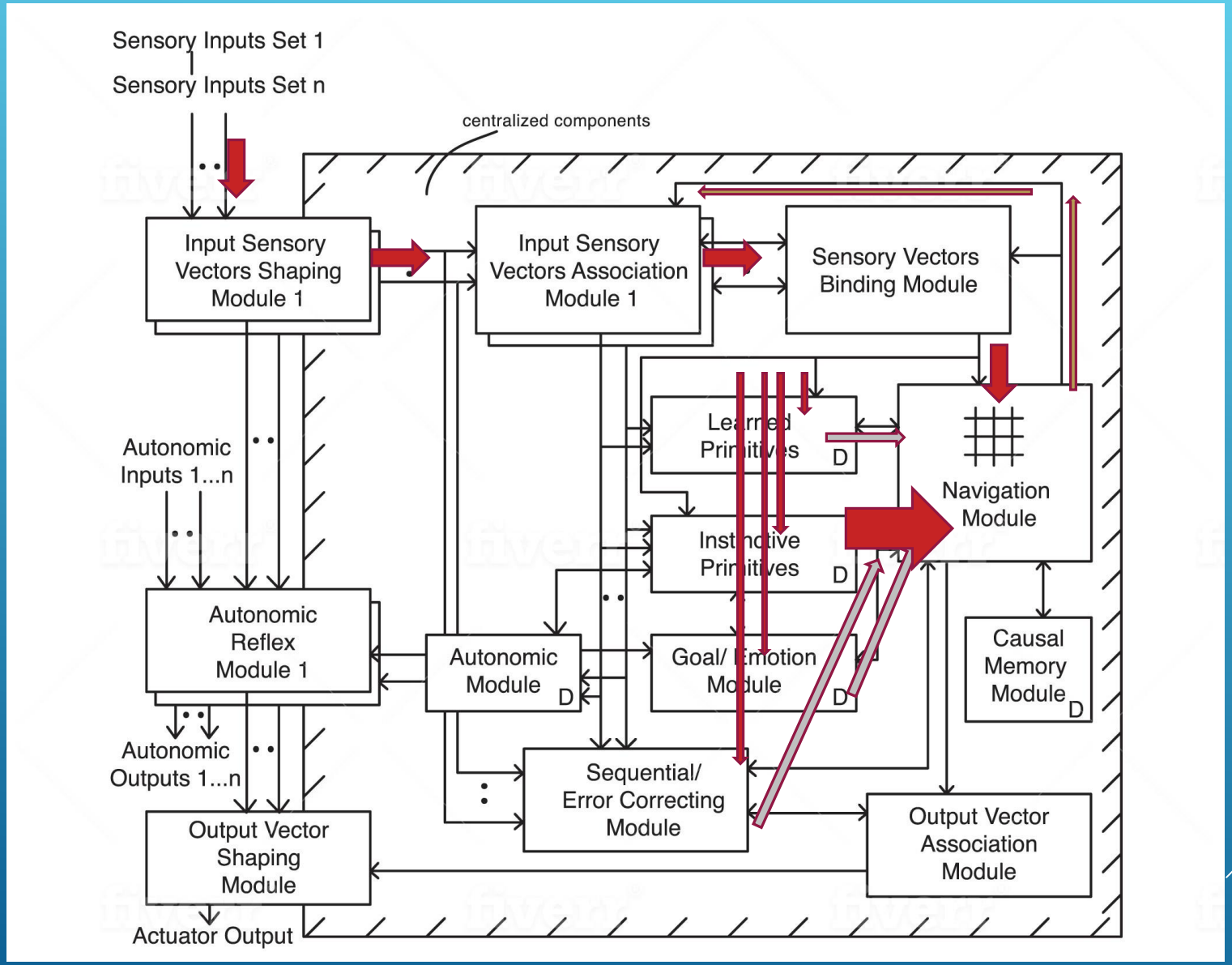
CCA1 – perception….



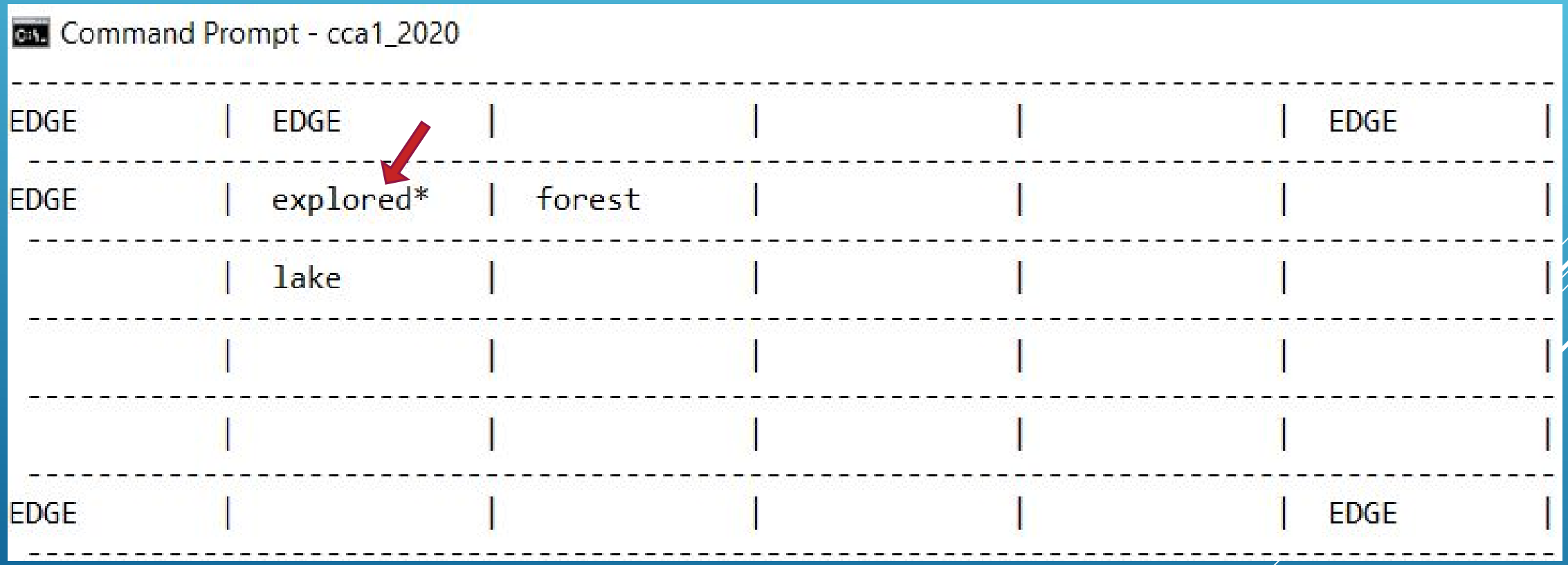
CCA1 builds up internal map from perceptions (and processing) in N, E, S, W directions



Lake (deep water) – Instinctive Primitive – do not go
Forest – Instinctive Primitive – no signal

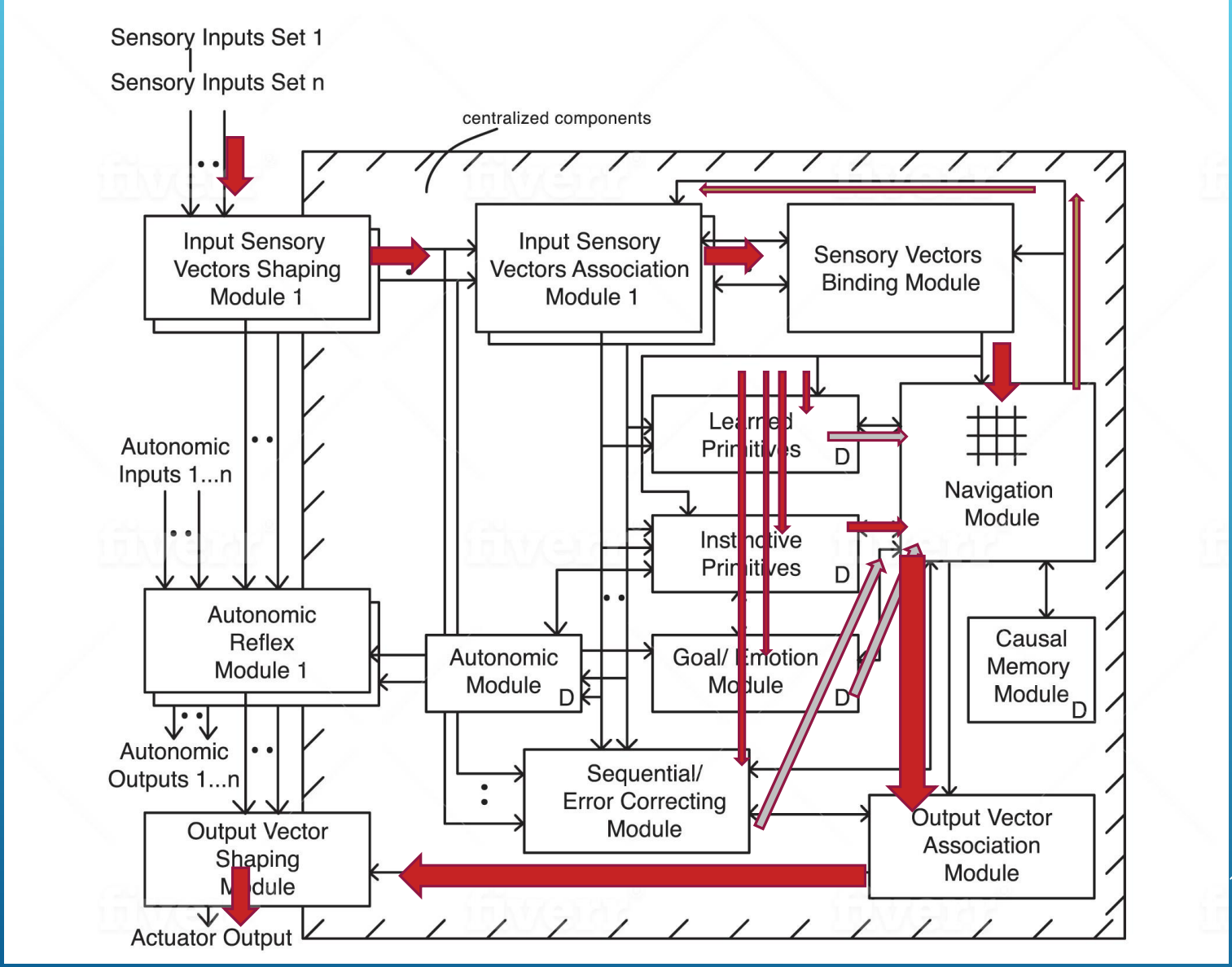


CCA1 builds up internal map from perceptions (and processing) in N, E, S, W directions




N
W
E
S

Navigation to the East (to the forest square)

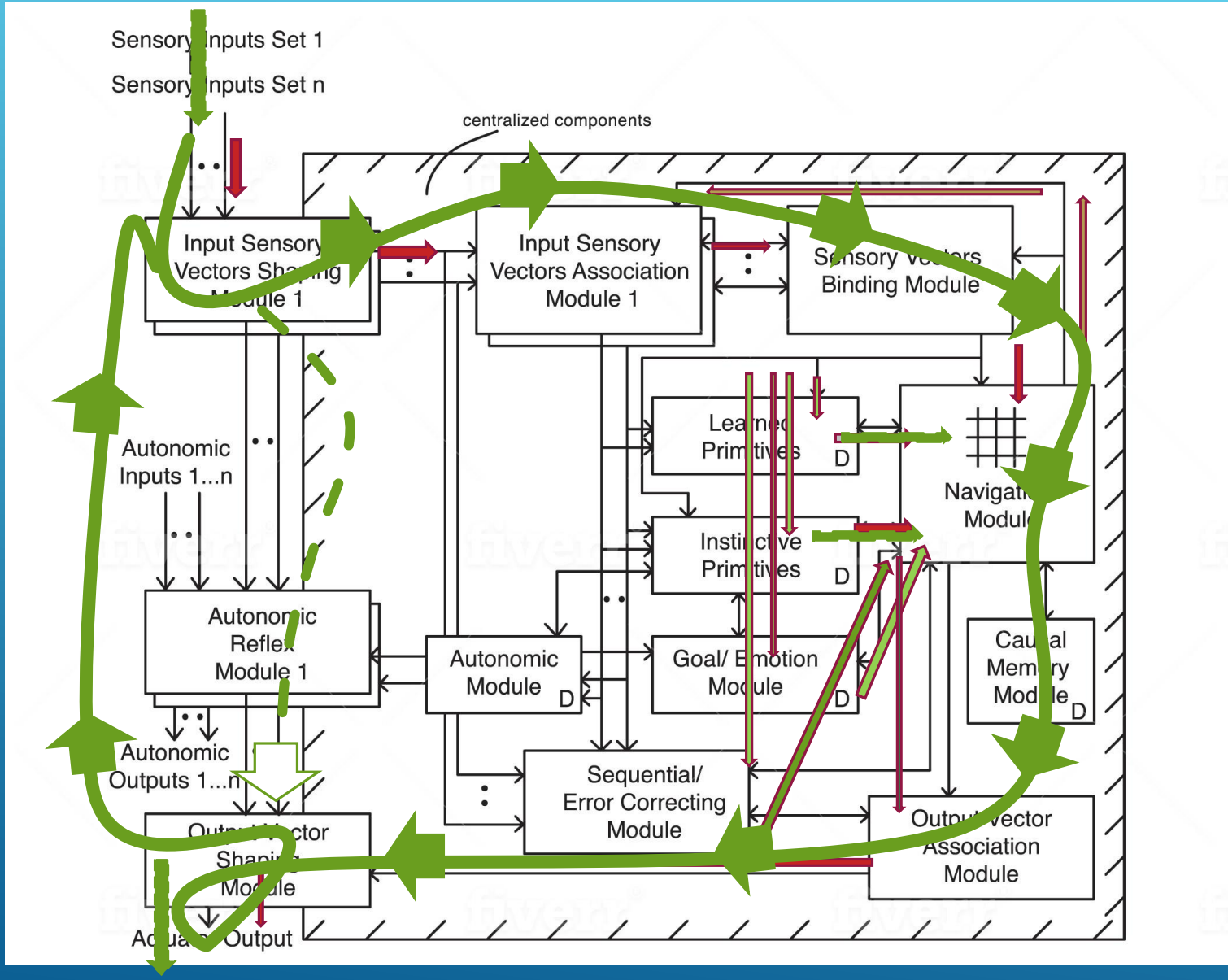


CCA1 moves East into 'forest' square

Command Prompt - cca1_2020

```
CCA1 moved from (1, 1)  1,2

Bird's-Eye View of Forest (CCA1 does not have this view)
-------------------------------------------------------------------------------
EDGE       |  EDGE       |  EDGE       |  EDGE       |  EDGE       |  EDGE       |
 ------------------------------------------------------------------------------
EDGE       |  forest     |  CCA1  *    |  sh_rvr     |  forest     |  EDGE       |
 ------------------------------------------------------------------------------
EDGE       |  lake       |  forest     |  forest     |  forest     |  EDGE       |
 ------------------------------------------------------------------------------
EDGE       |  forest     |  wtrfall    |  forest     |  forest     |  EDGE       |
 ------------------------------------------------------------------------------
EDGE       |  forest     |  hiker      |  forest     |  forest     |  EDGE       |
 ------------------------------------------------------------------------------
EDGE       |  EDGE       |  EDGE       |  EDGE       |  EDGE       |  EDGE       |
```

"Processing Cycles" repeat over and over again



No Special Central Controlling Stored Program

No computer-like clock circuitry centrally controlling CCA1

Vectors propagated from circuit to circuit, and then the cycle is repeated

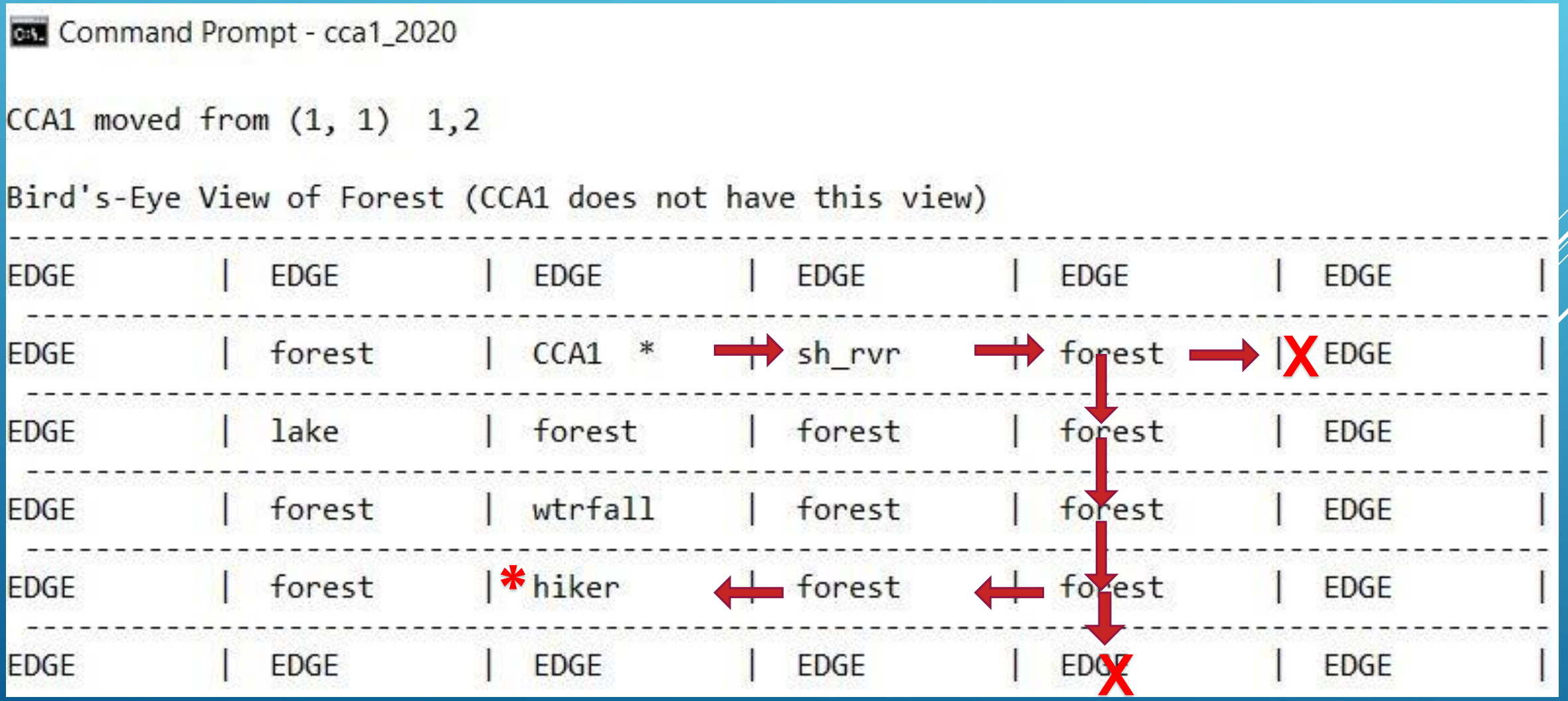
CCA1 must navigate to the lost hiker's square

Command Prompt - cca1_2020

```
hiker position set to:  4 2

Bird's-Eye View of Forest (CCA1 does not have this view)
```

EDGE	EDGE	EDGE	EDGE	EDGE	EDGE
EDGE	CCA1 *	forest	sh_rvr	forest	EDGE
EDGE	lake	forest	forest	forest	EDGE
EDGE	forest	wtrfall	forest	forest	EDGE
EDGE	forest	hiker	forest	forest	EDGE
EDGE	EDGE	EDGE	EDGE	EDGE	EDGE

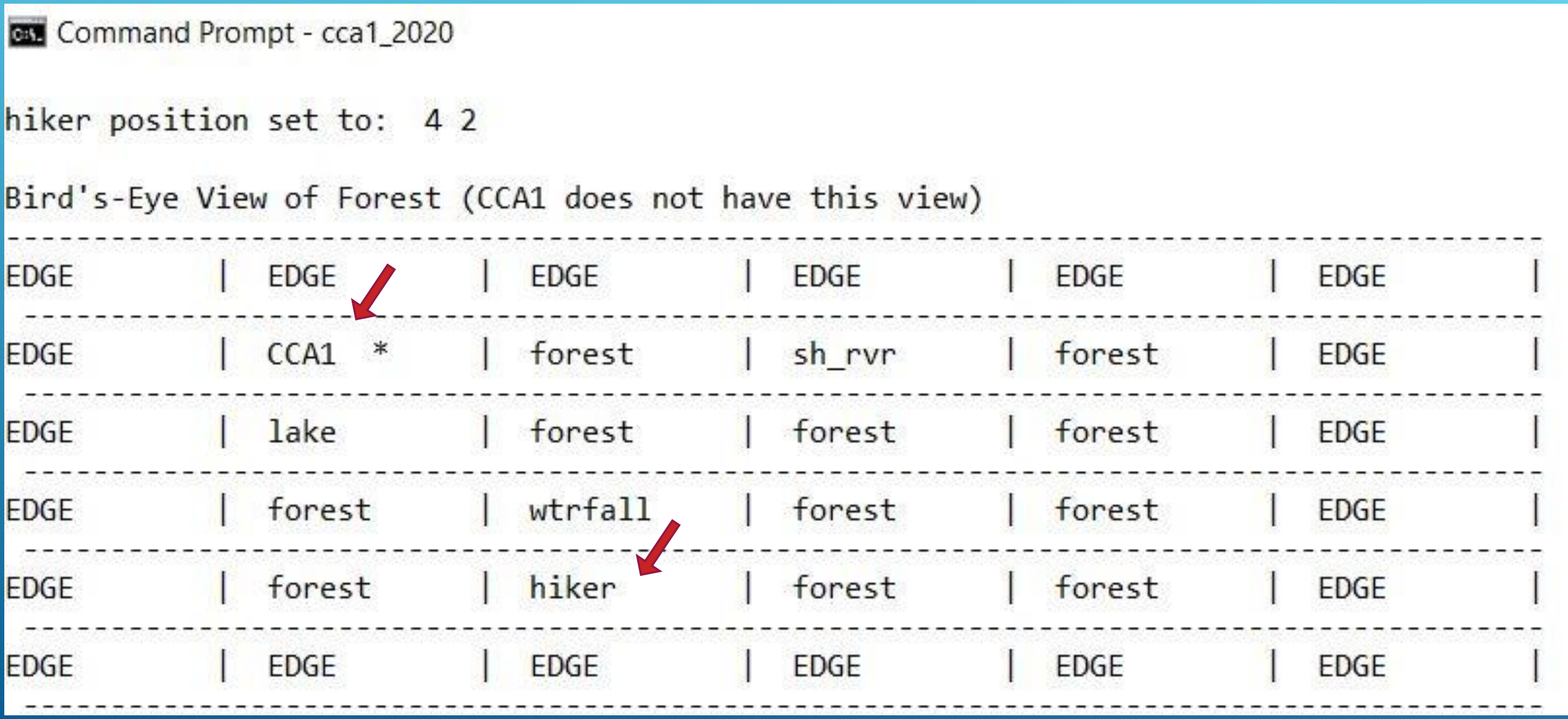
CCA1 eventually navigates to the hiker square, and rescues the lost hiker



Start new CCA1 simulation….

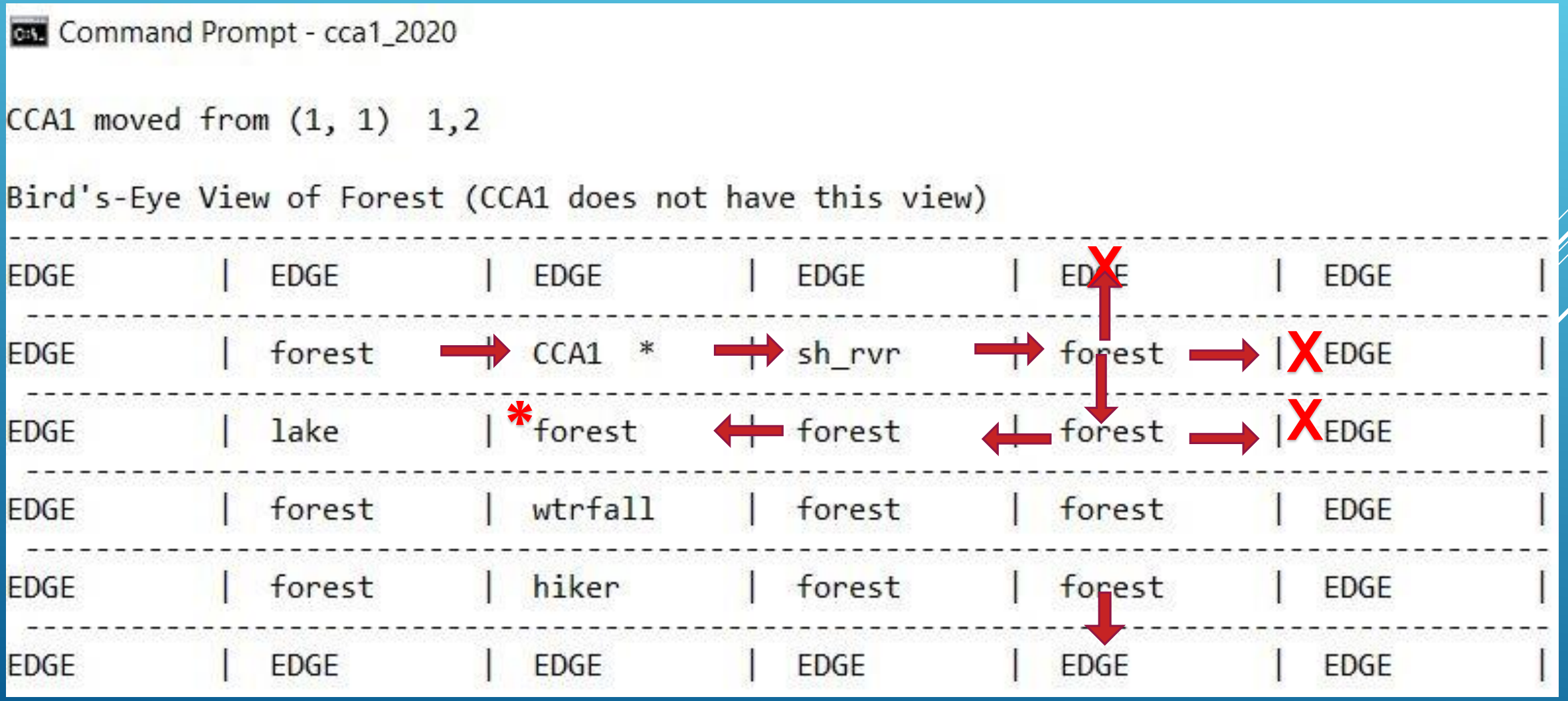
Command Prompt - cca1_2020

```
hiker position set to:  4 2

Bird's-Eye View of Forest (CCA1 does not have this view)
```



EDGE	EDGE	EDGE	EDGE	EDGE	EDGE
EDGE	CCA1 *	forest	sh_rvr	forest	EDGE
EDGE	lake	forest	forest	forest	EDGE
EDGE	forest	wtrfall	forest	forest	EDGE
EDGE	forest	hiker	forest	forest	EDGE
EDGE	EDGE	EDGE	EDGE	EDGE	EDGE

CCA1 moves to north of the waterfall square….

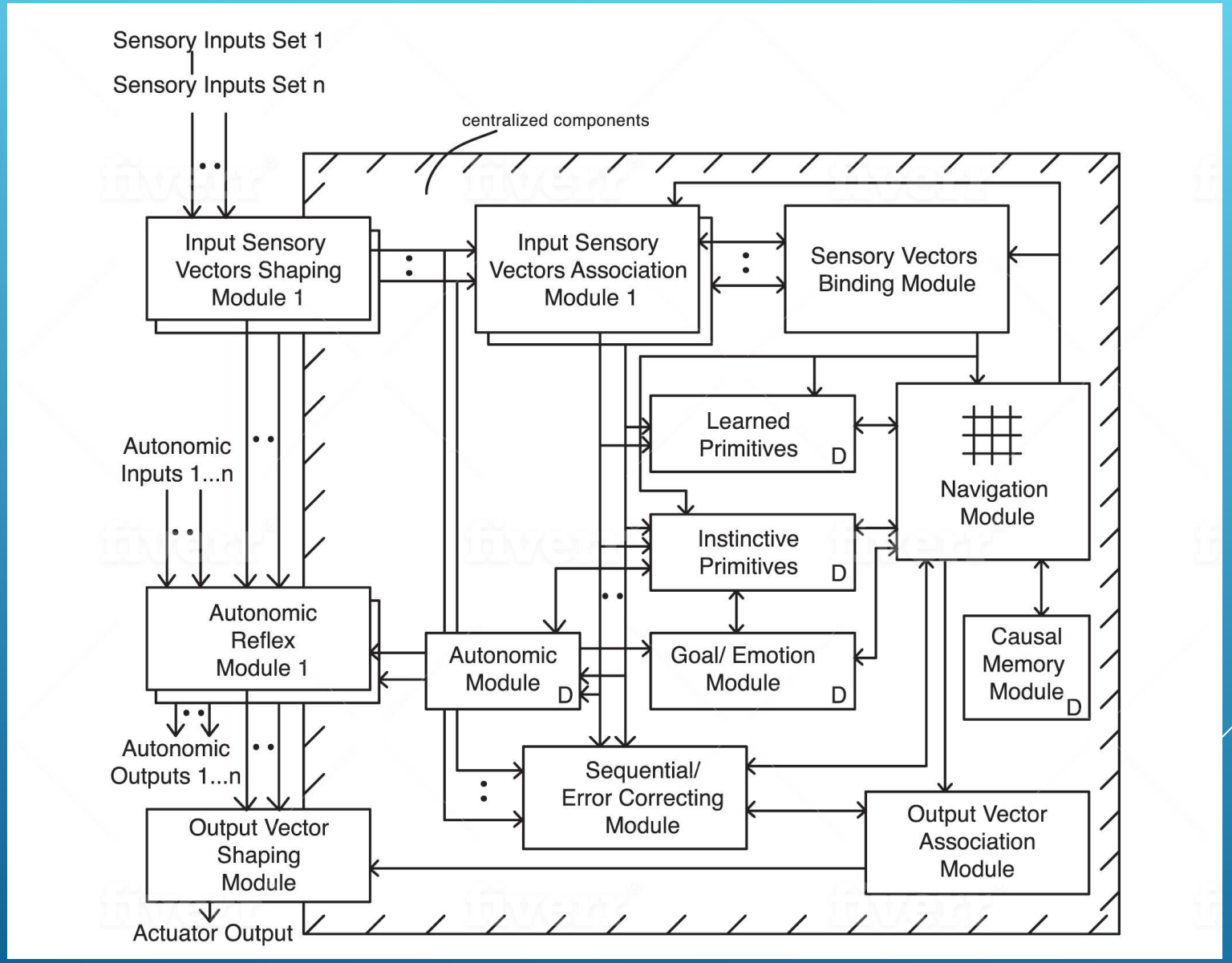


CCA1 has moved north of the waterfall square…

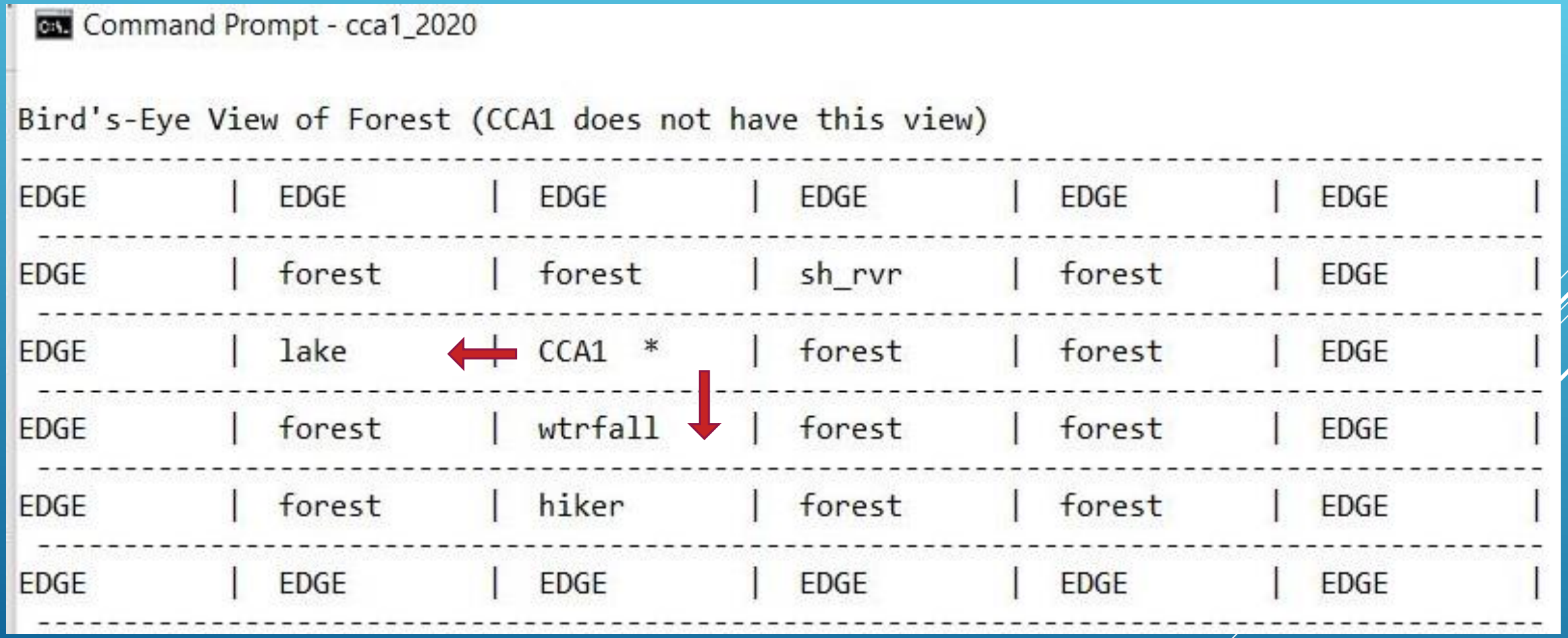
Command Prompt - cca1_2020

```
Bird's-Eye View of Forest (CCA1 does not have this view)
-------------------------------------------------------------------------------
EDGE       | EDGE       | EDGE       | EDGE       | EDGE       | EDGE       |
-------------------------------------------------------------------------------
EDGE       | forest     | forest     | sh_rvr     | forest     | EDGE       |
-------------------------------------------------------------------------------
EDGE       | lake       | CCA1  *    | forest     | forest     | EDGE       |
-------------------------------------------------------------------------------
EDGE       | forest     | wtrfall    | forest     | forest     | EDGE       |
-------------------------------------------------------------------------------
EDGE       | forest     | hiker      | forest     | forest     | EDGE       |
-------------------------------------------------------------------------------
EDGE       | EDGE       | EDGE       | EDGE       | EDGE       | EDGE       |
-------------------------------------------------------------------------------
```

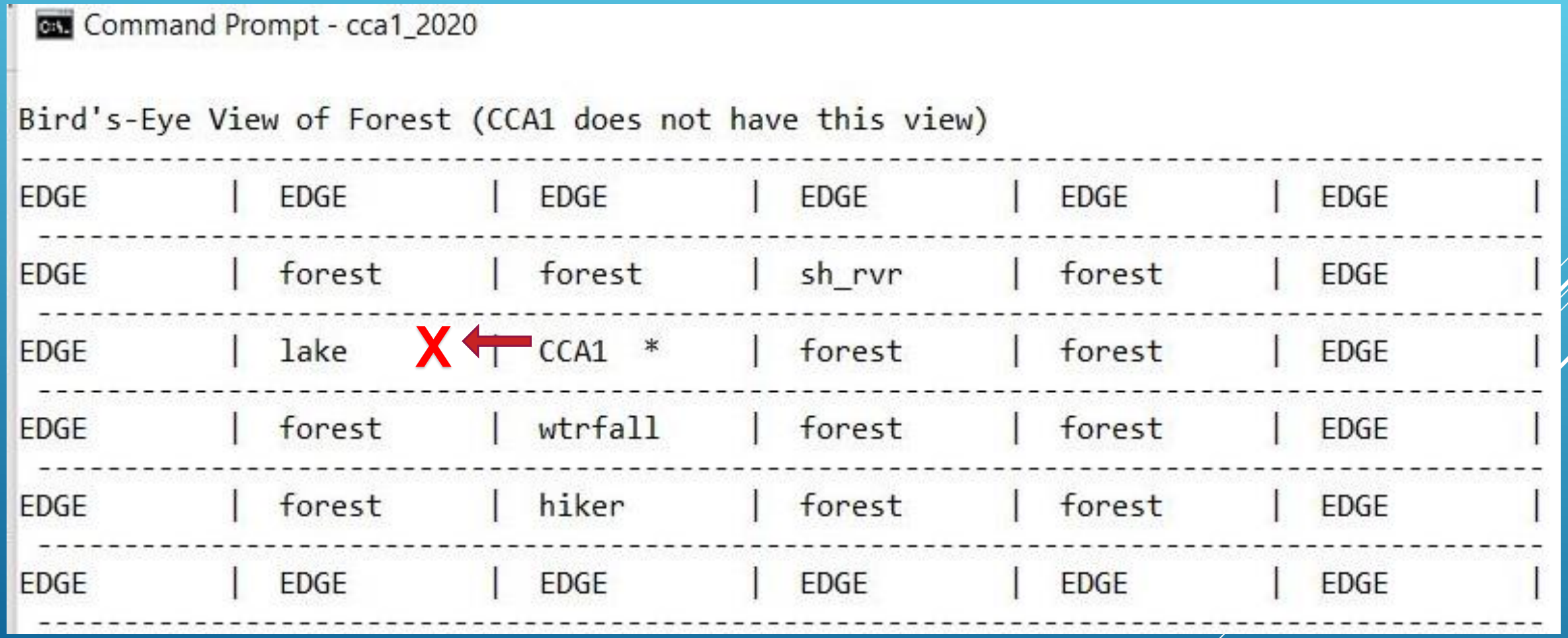
Causal Cognitive Architecture 1 (CCA1)



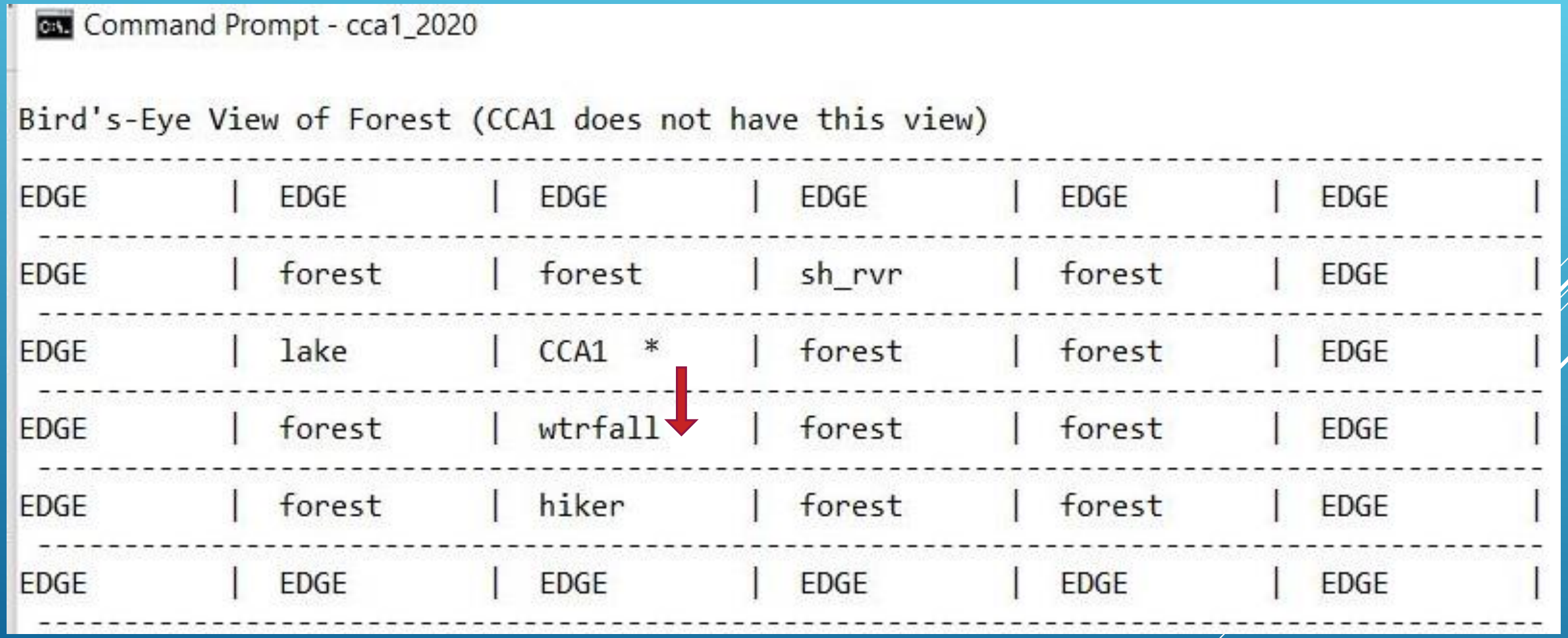
Goal/Emotion Module favors trying W & S to find the lost hiker



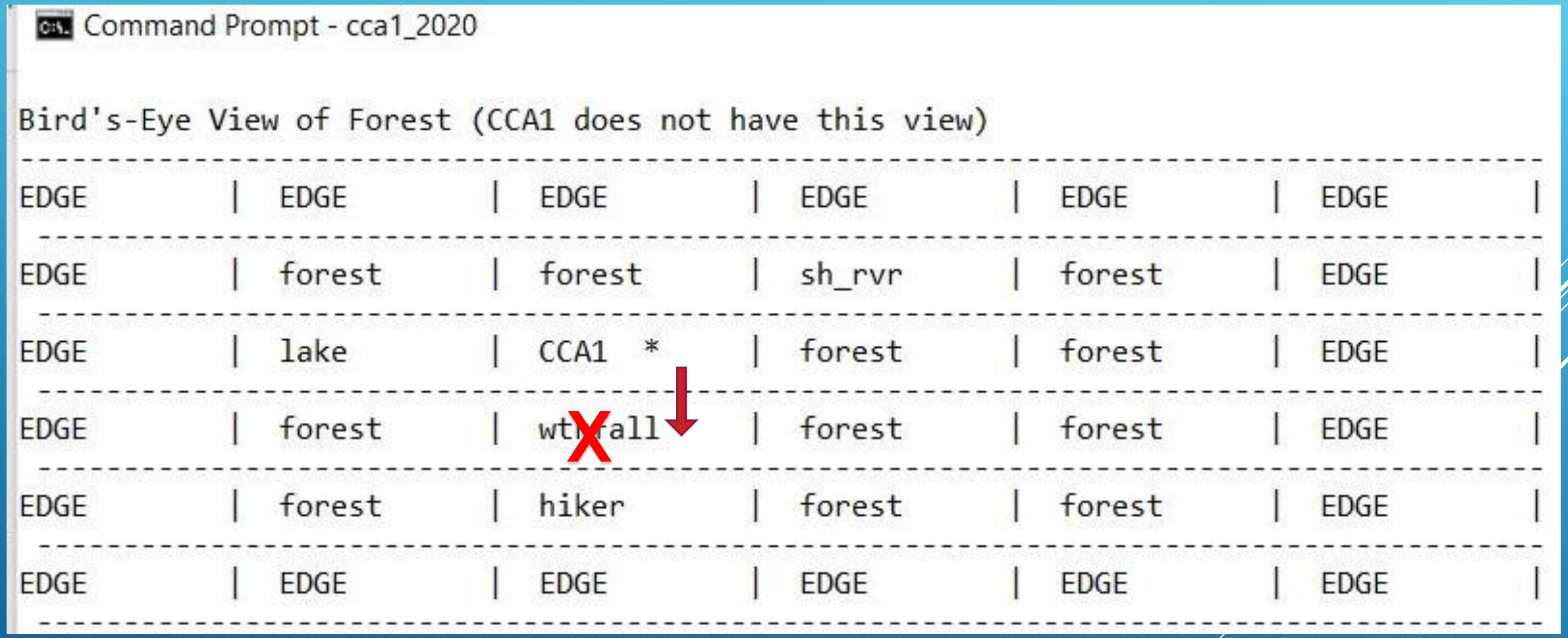
Instinctive Primitives rejects move W to lake



S – sees fast noisy river (does not see cliff part) Able to cross shallow rivers, so moves South



CCA1 moves S (south) and is swept off cliff of waterfall and is damaged – mission ends



Associative Learning Occurs

-Repair damaged CCA1

-Next day it goes out into the forest on another mission

-Sees fast flowing river with much noise

-Triggers in Goal/Emotion Module and Learned Primitives Module not to go there

-Makes another choice for direction of move

-New Simulation
-Use full causal features of architecture

Command Prompt - cca1_2020

```
Please choose type of "hippocampus"/"brain" which, of course,
 only loosely approximates the biological equivalent:
1. Lamprey hippocampal/brain analogue
2. Fish hippocampal/telencephalon analogue
3. Reptile hippocampal/pallium analogue
4. Mammalian hippocampus - note: meaningfulness, precausal
5. Human hippocampus - note: meaningfulness plus full causal features
6. Augmented Human level 1 - simultaneous multiple navigational threads
7. Augmented Human level 2 - algorithm center in each navigational module
Please make a selection:
```

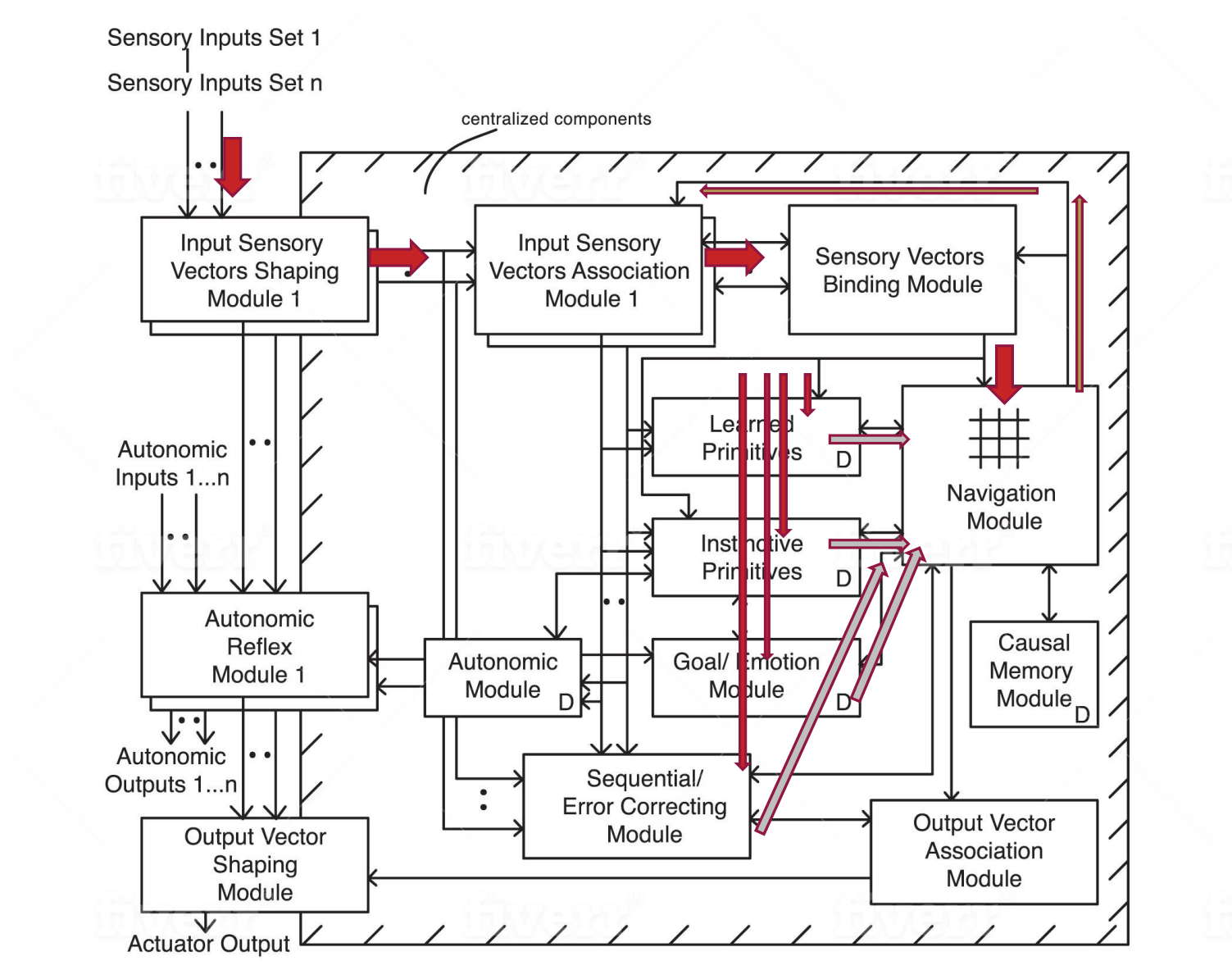
CCA1 must navigate to the lost hiker's square

Command Prompt - cca1_2020

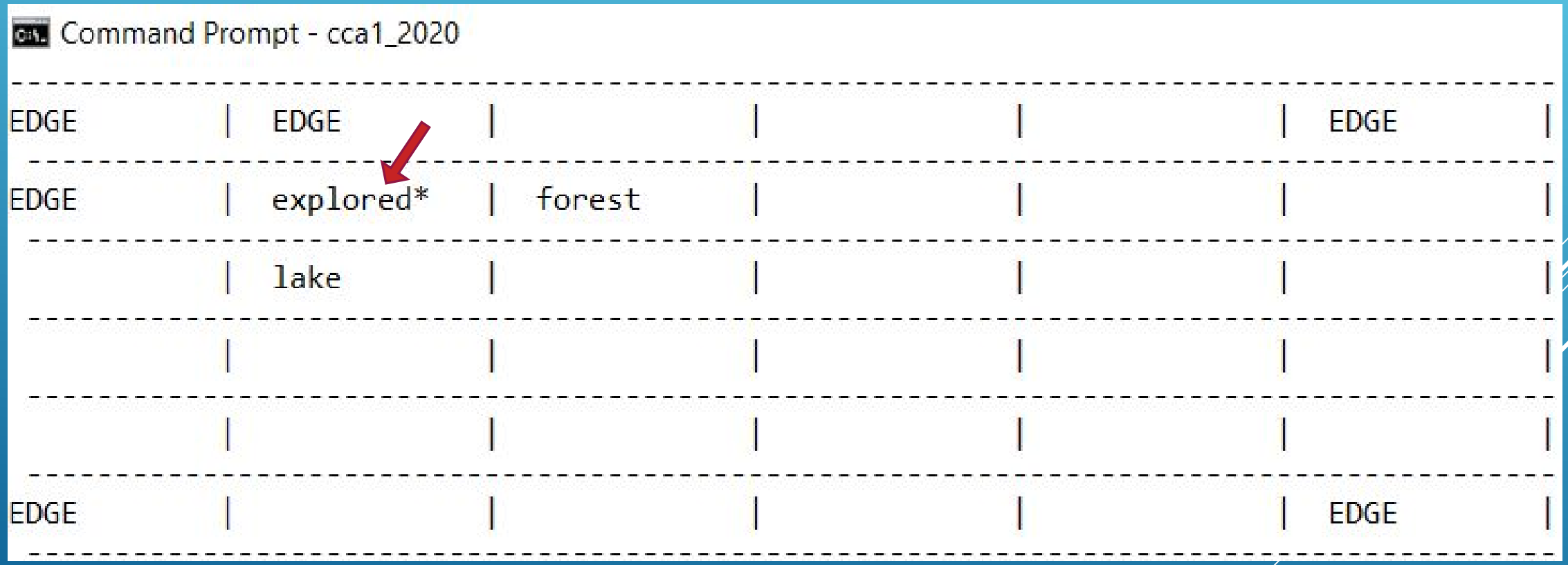
```
hiker position set to:  4 2

Bird's-Eye View of Forest (CCA1 does not have this view)
```

EDGE	EDGE	EDGE	EDGE	EDGE	EDGE
EDGE	CCA1 *	forest	sh_rvr	forest	EDGE
EDGE	lake	forest	forest	forest	EDGE
EDGE	forest	wtrfall	forest	forest	EDGE
EDGE	forest	hiker	forest	forest	EDGE
EDGE	EDGE	EDGE	EDGE	EDGE	EDGE

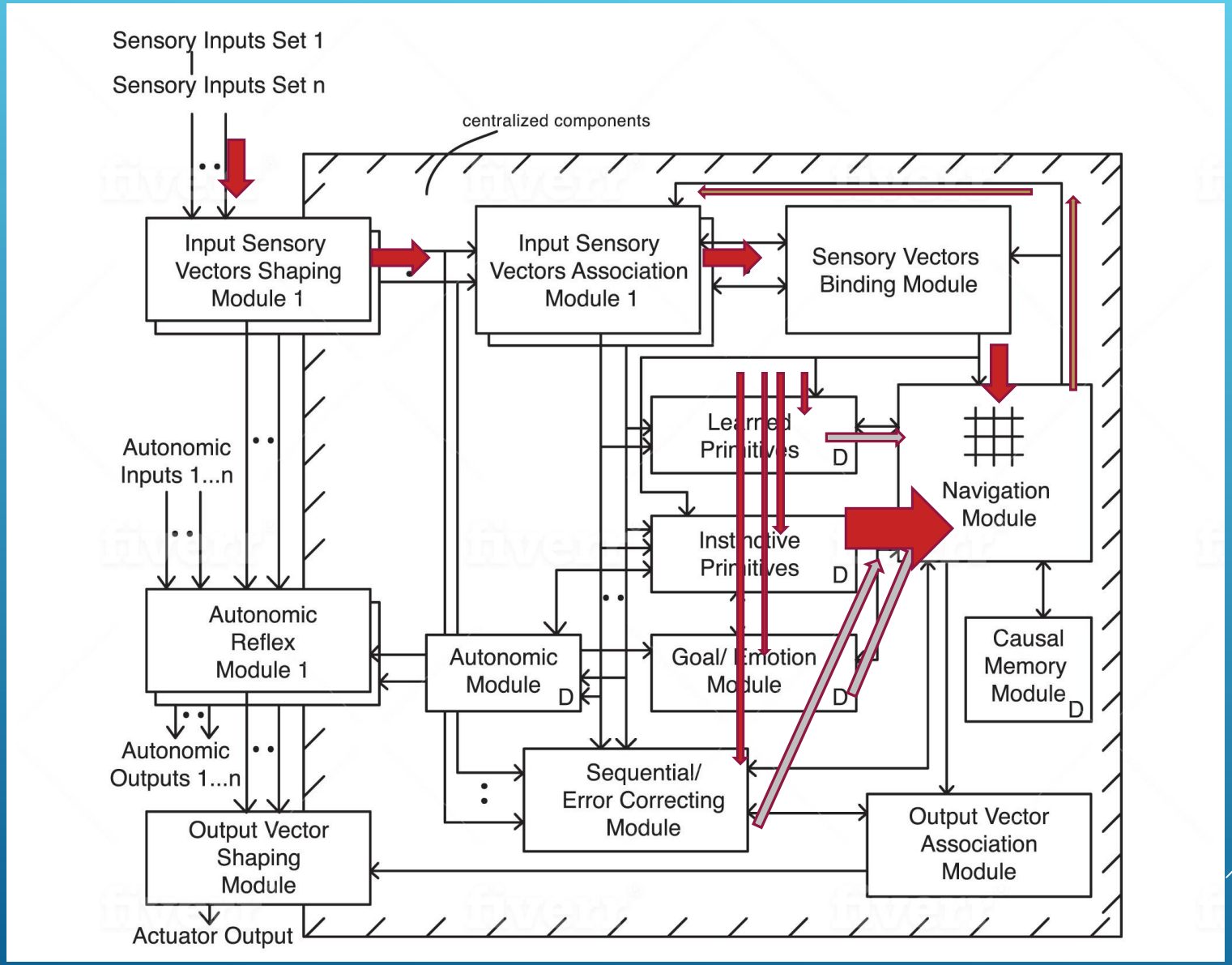
CCA1 – perception….



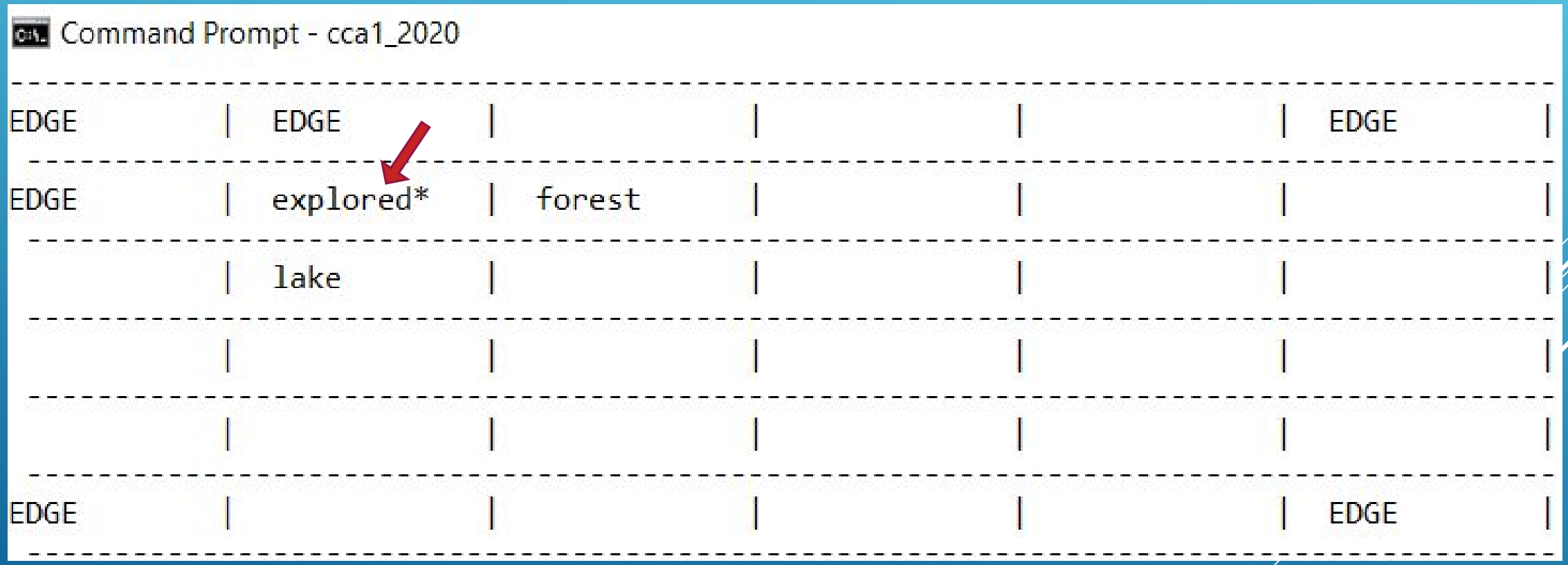
CCA1 builds up internal map from perceptions (and processing) in N, E, S, W directions



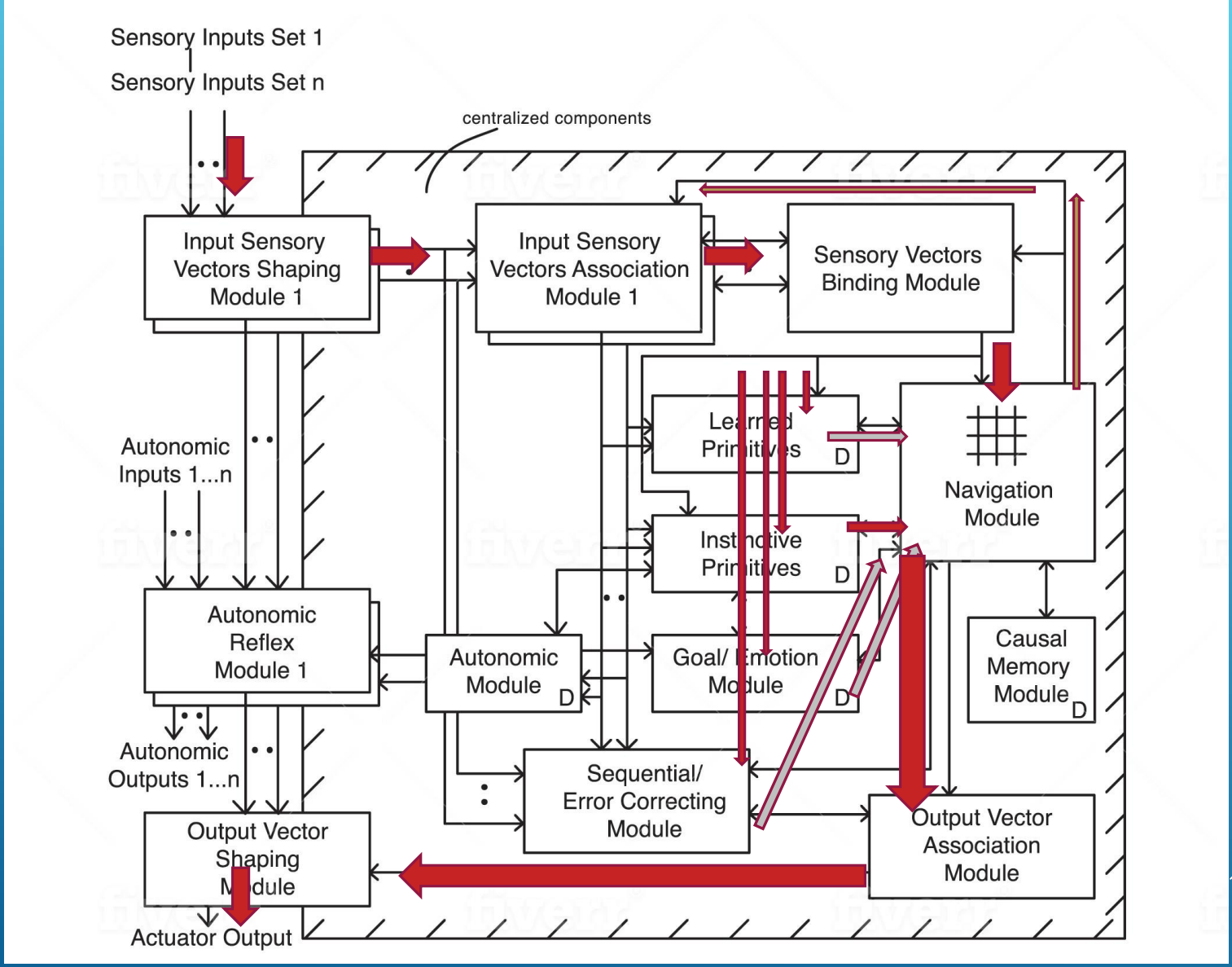
Lake (deep water) – Instinctive Primitive – do not go
Forest – Instinctive Primitive – no signal



CCA1 builds up internal map from perceptions (and processing) in N, E, S, W directions



Navigation to the East (to the forest square)

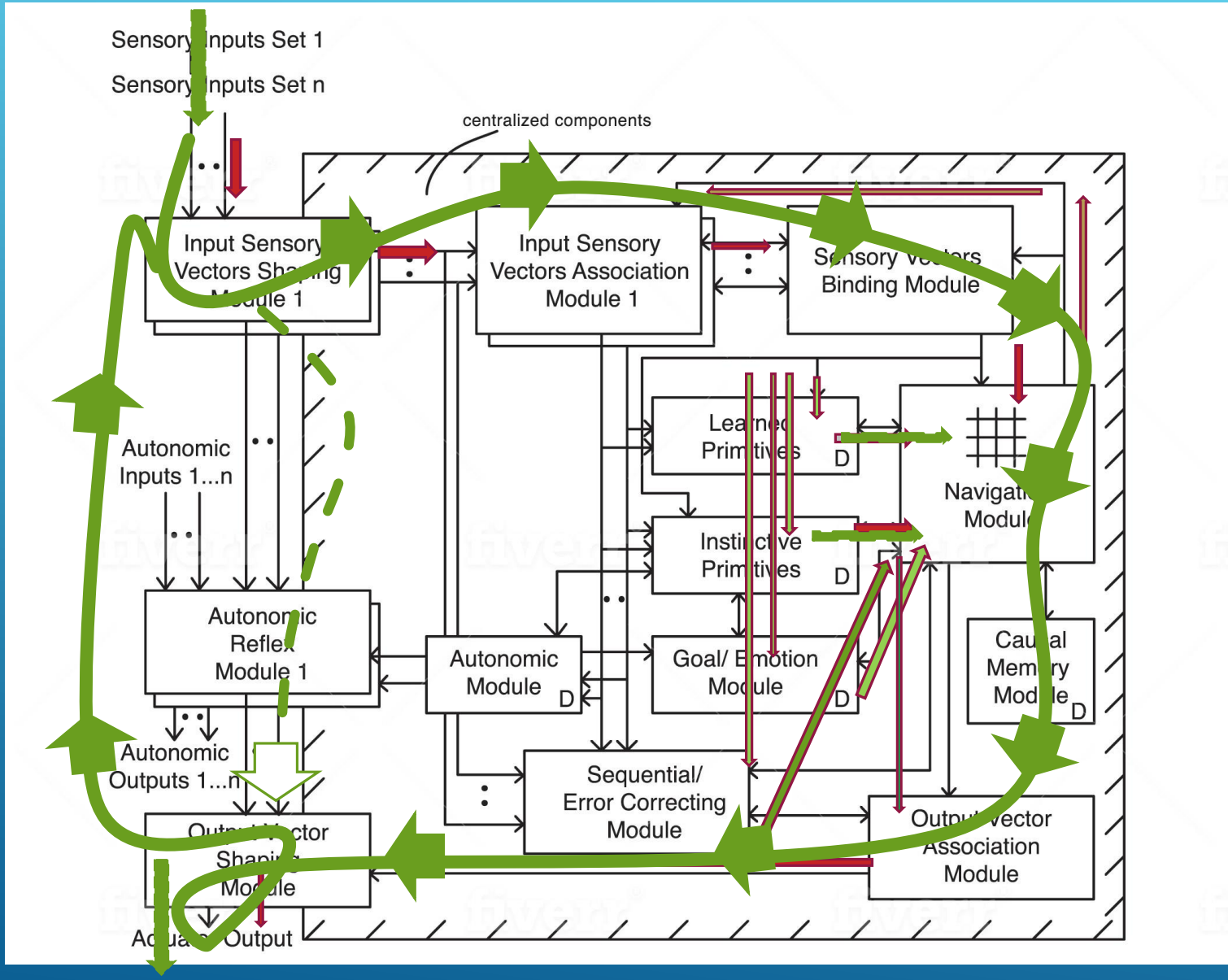


CCA1 moves East into 'forest' square

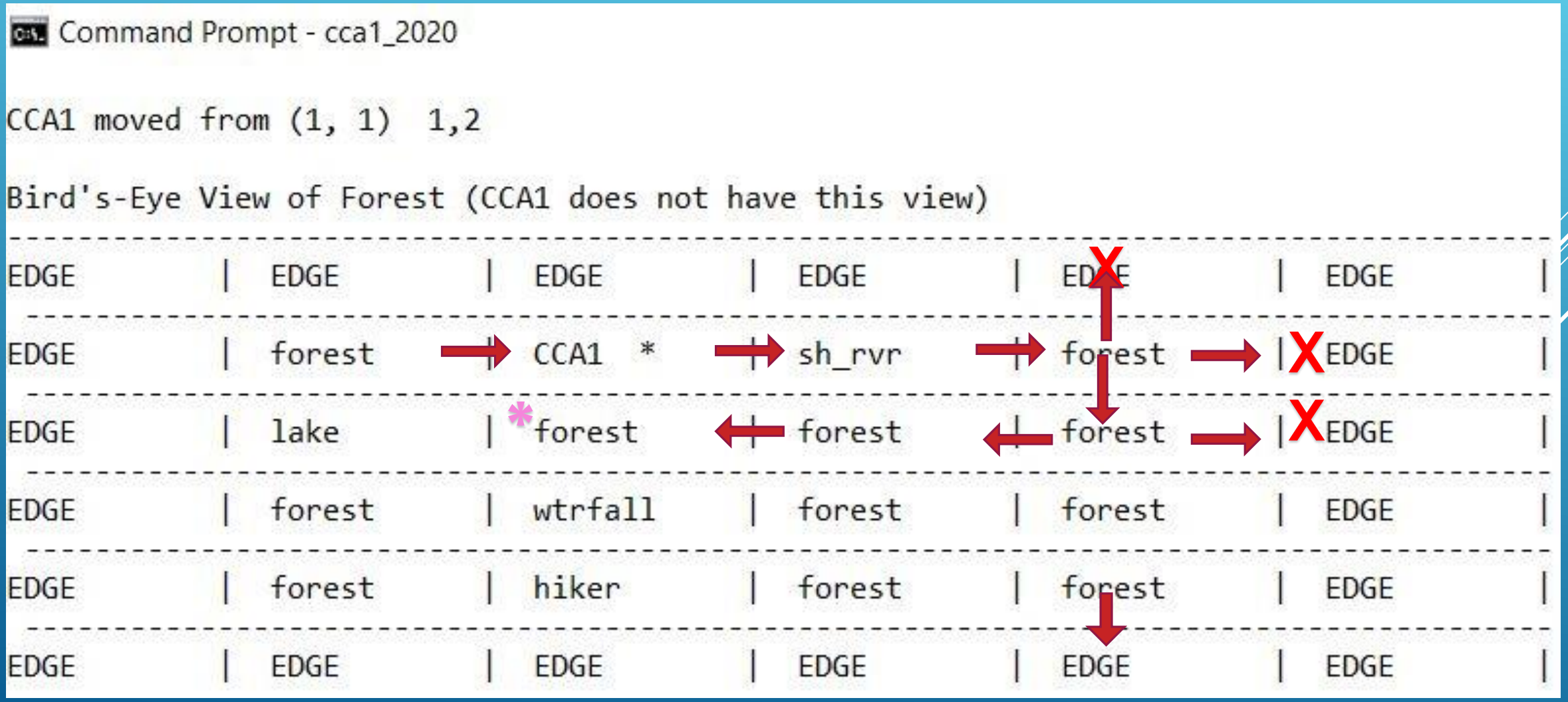
Command Prompt - cca1_2020

```
CCA1 moved from (1, 1)  1,2

Bird's-Eye View of Forest (CCA1 does not have this view)
-------------------------------------------------------------------------------
EDGE       |  EDGE       |  EDGE       |  EDGE       |  EDGE       |  EDGE       |
 ------------------------------------------------------------------------------
EDGE       |  forest     |  CCA1  *    |  sh_rvr     |  forest     |  EDGE       |
 ------------------------------------------------------------------------------
EDGE       |  lake       |  forest     |  forest     |  forest     |  EDGE       |
 ------------------------------------------------------------------------------
EDGE       |  forest     |  wtrfall    |  forest     |  forest     |  EDGE       |
 ------------------------------------------------------------------------------
EDGE       |  forest     |  hiker      |  forest     |  forest     |  EDGE       |
 ------------------------------------------------------------------------------
EDGE       |  EDGE       |  EDGE       |  EDGE       |  EDGE       |  EDGE       |
```

"Processing Cycles" repeat over and over again



CCA1 moves to north of the waterfall square….

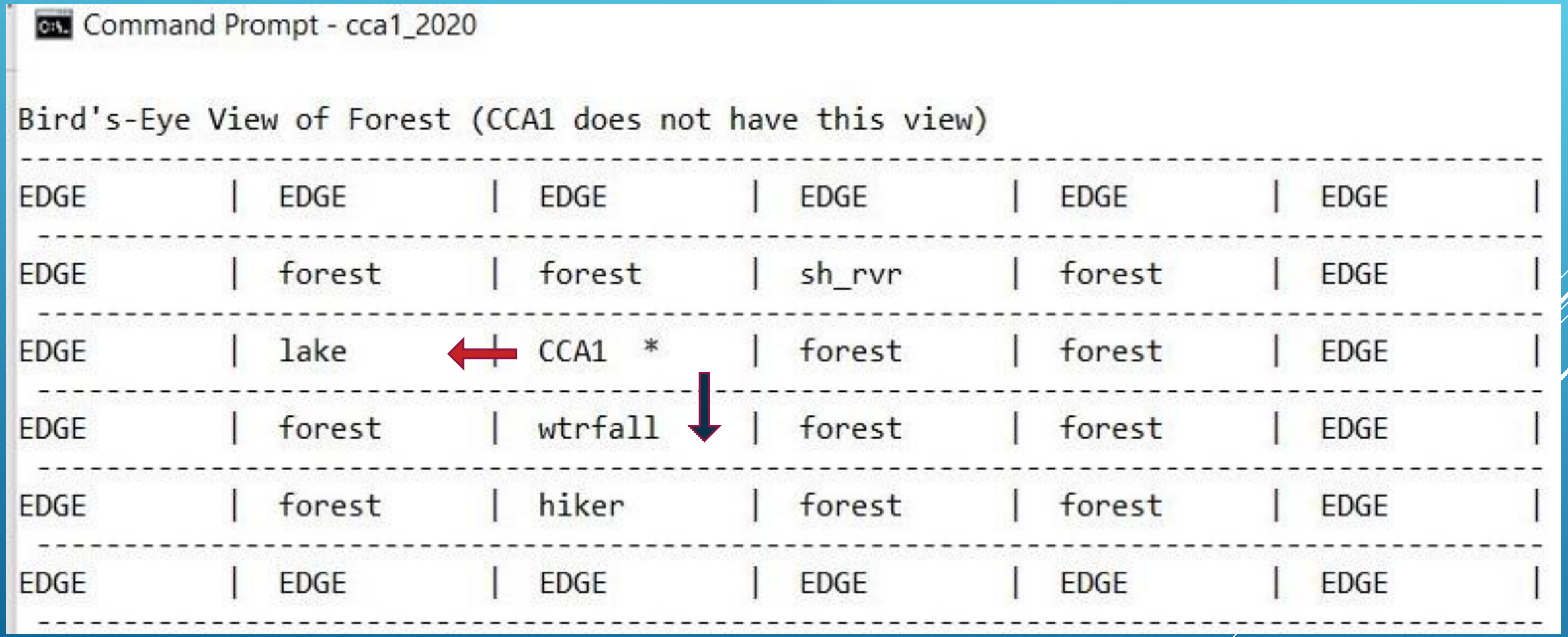


CCA1 has moved north of the waterfall square…

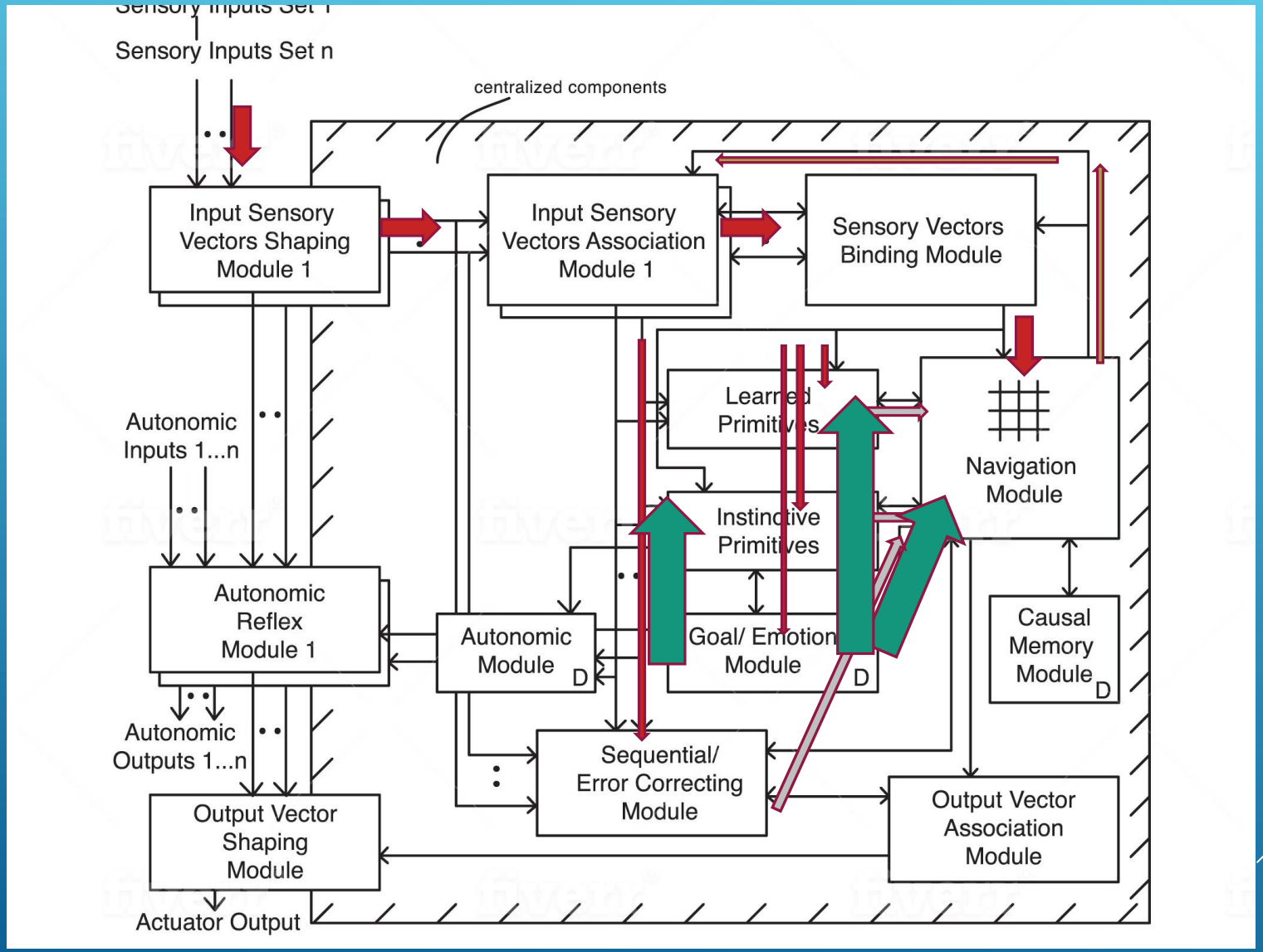
Command Prompt - cca1_2020

```
Bird's-Eye View of Forest (CCA1 does not have this view)
-------------------------------------------------------------------------------
EDGE       | EDGE       | EDGE       | EDGE       | EDGE       | EDGE       |
-------------------------------------------------------------------------------
EDGE       | forest     | forest     | sh_rvr     | forest     | EDGE       |
-------------------------------------------------------------------------------
EDGE       | lake       | CCA1  *    | forest     | forest     | EDGE       |
-------------------------------------------------------------------------------
EDGE       | forest     | wtrfall    | forest     | forest     | EDGE       |
-------------------------------------------------------------------------------
EDGE       | forest     | hiker      | forest     | forest     | EDGE       |
-------------------------------------------------------------------------------
EDGE       | EDGE       | EDGE       | EDGE       | EDGE       | EDGE       |
-------------------------------------------------------------------------------
```

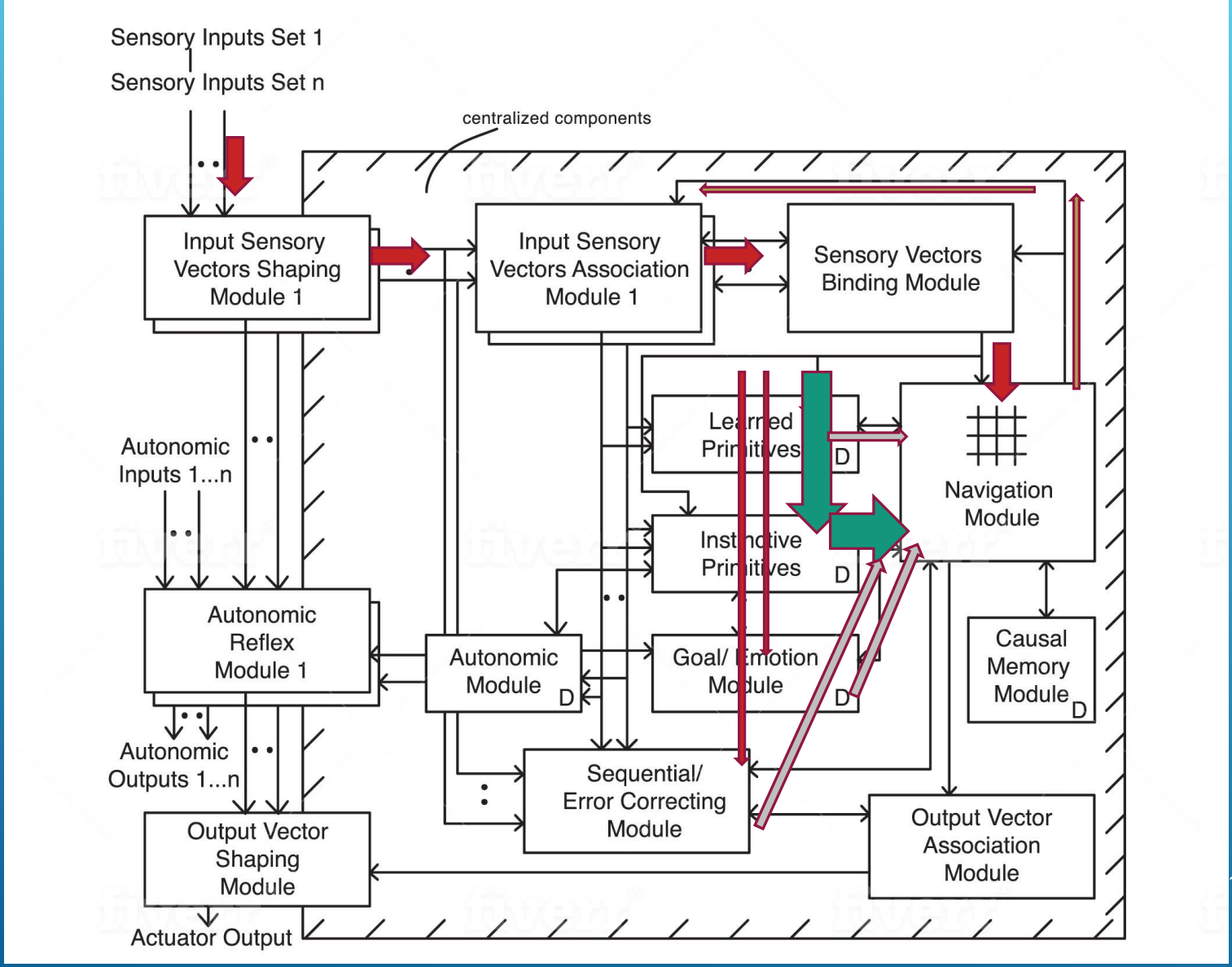
-Goal/Emotion Module favors trying W & S
-Ok to move W? (Then consider ok to move S?)



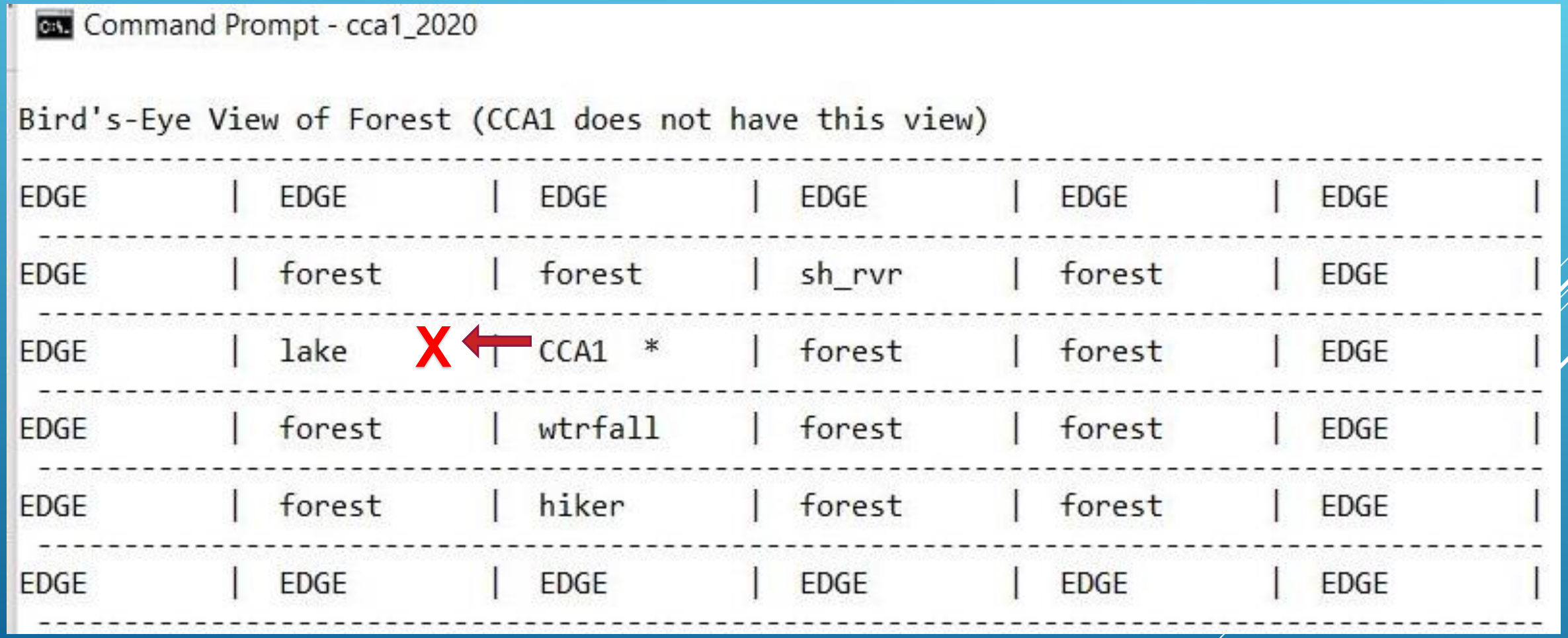
Goal/Emotion Module will Affect Primitives Chosen as well as operations of Navigation Module



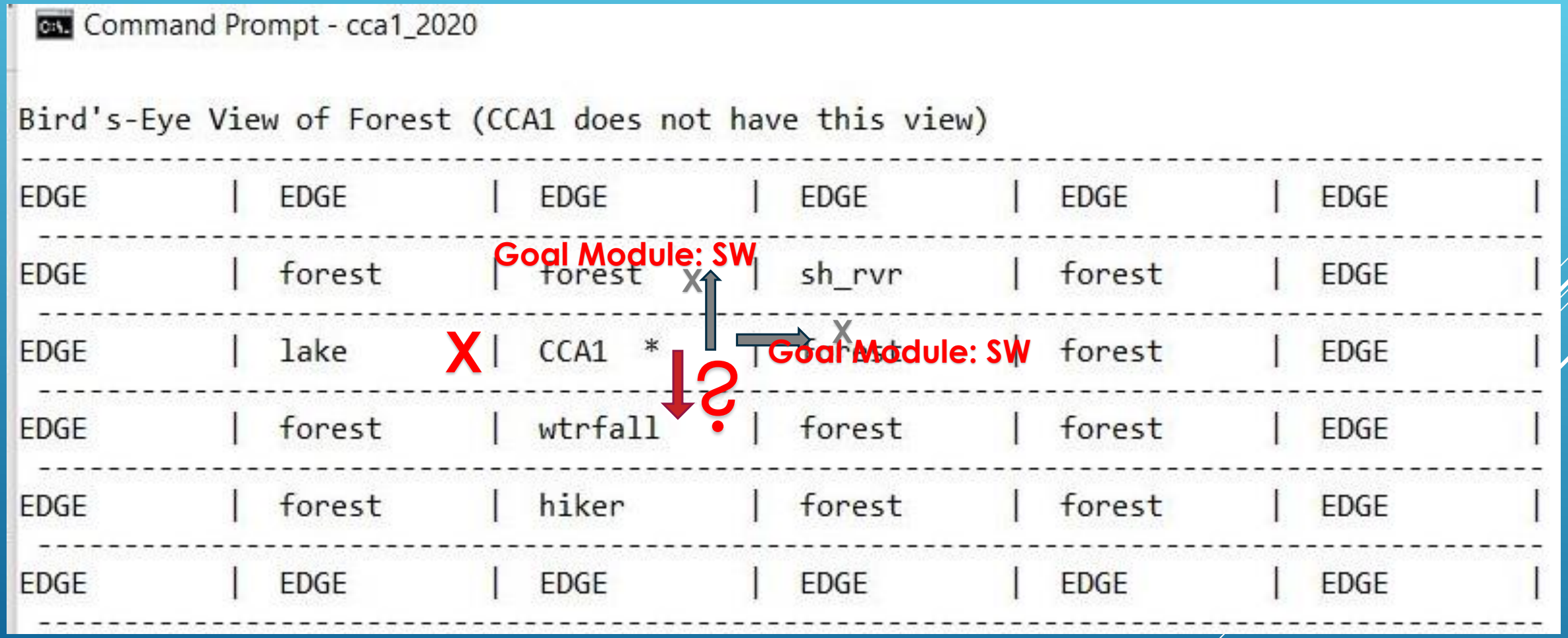
Instinctive Primitives rejects move W to lake



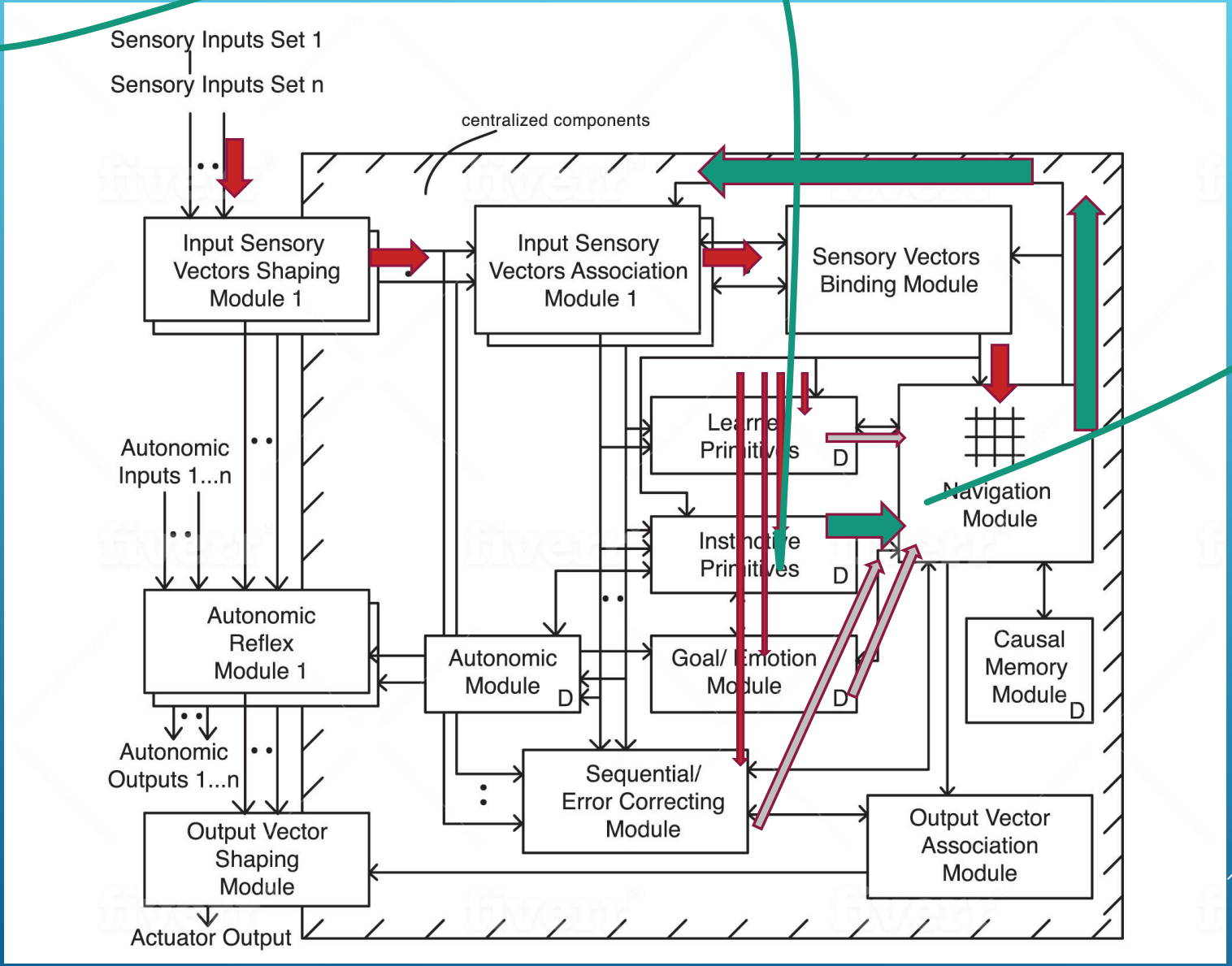
Instinctive Primitives rejects move W to lake



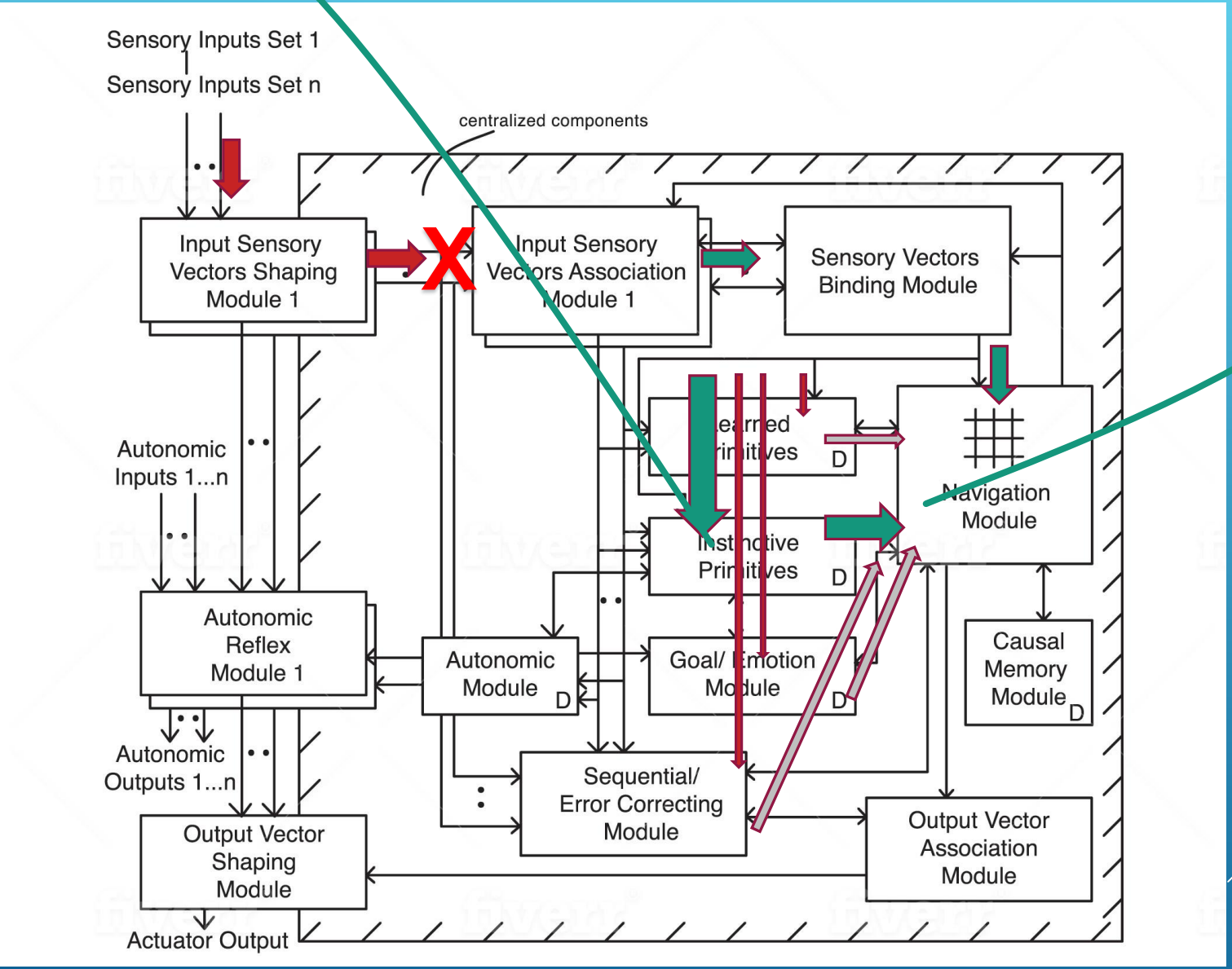
S – sees fast noisy river (does not see cliff part)



{“water”} + {“fast flow” + “noise”} → {“water” + “push”}



{“water” + “push”} → triggers new map in Navig’nModule

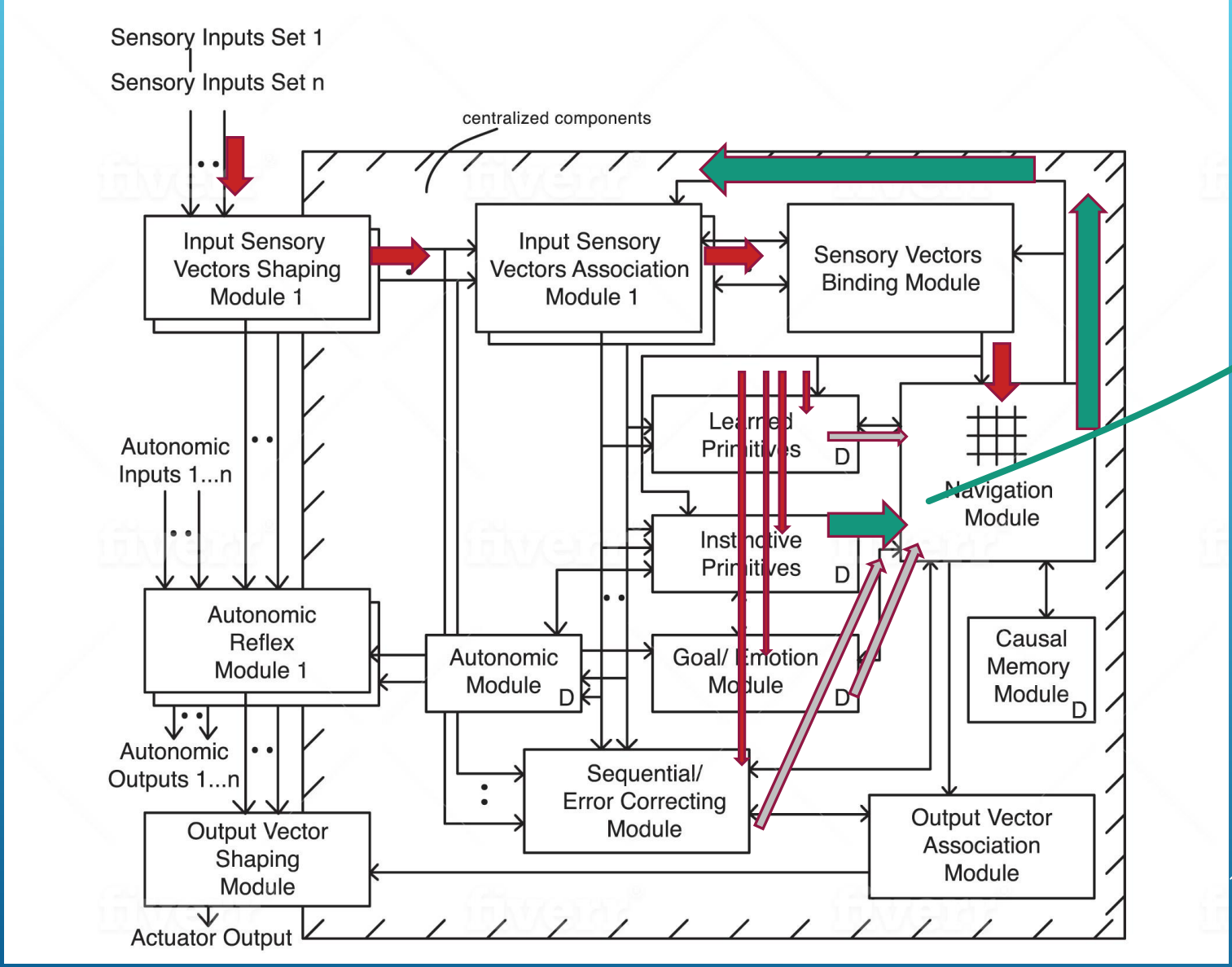


Temporary map → {“CCA1 under water”}

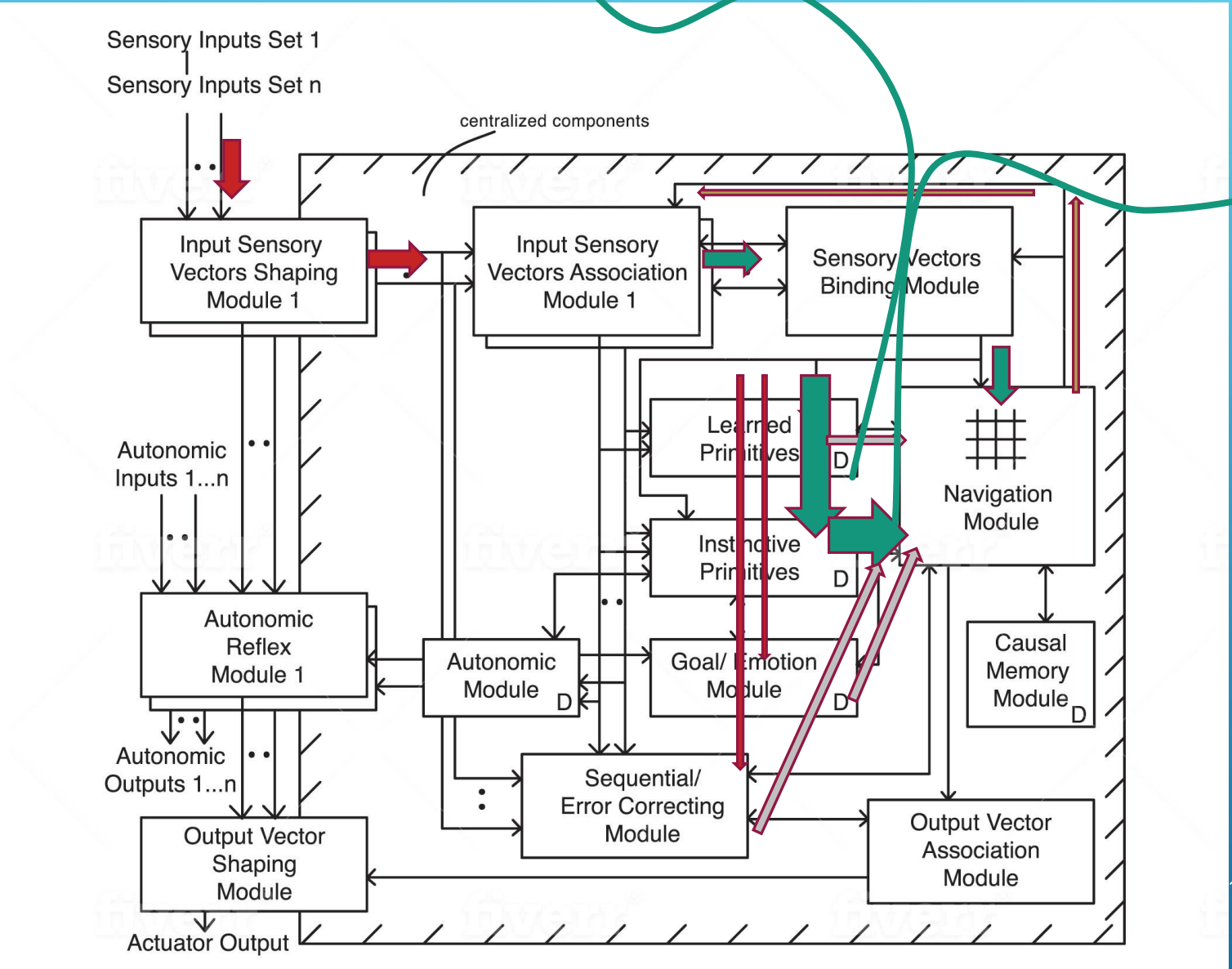
Command Prompt - cca1_2020

```
Internal Map From Stack
-------------------------------------------------------------------------------
air     | air     | air     | air     | air     | air     |
 -------------------------------------------------------------------------------
water   | water   | water   | water   | water   | water   |
 -------------------------------------------------------------------------------
water   | water   | water   | water   | water   | water   |
 -------------------------------------------------------------------------------
water   | water   | water   | CCA1  *   | water   | water   |
 -------------------------------------------------------------------------------
water   | water   | water   | water   | water   | water   |
 -------------------------------------------------------------------------------
water   | water   | water   | water   | water   | water   |
 -------------------------------------------------------------------------------
water   | water   | water   | water   | water   | water   |
 -------------------------------------------------------------------------------
```

{“CCA1 under water”} is fed back to sensory input module



{“CCA1 under water”}

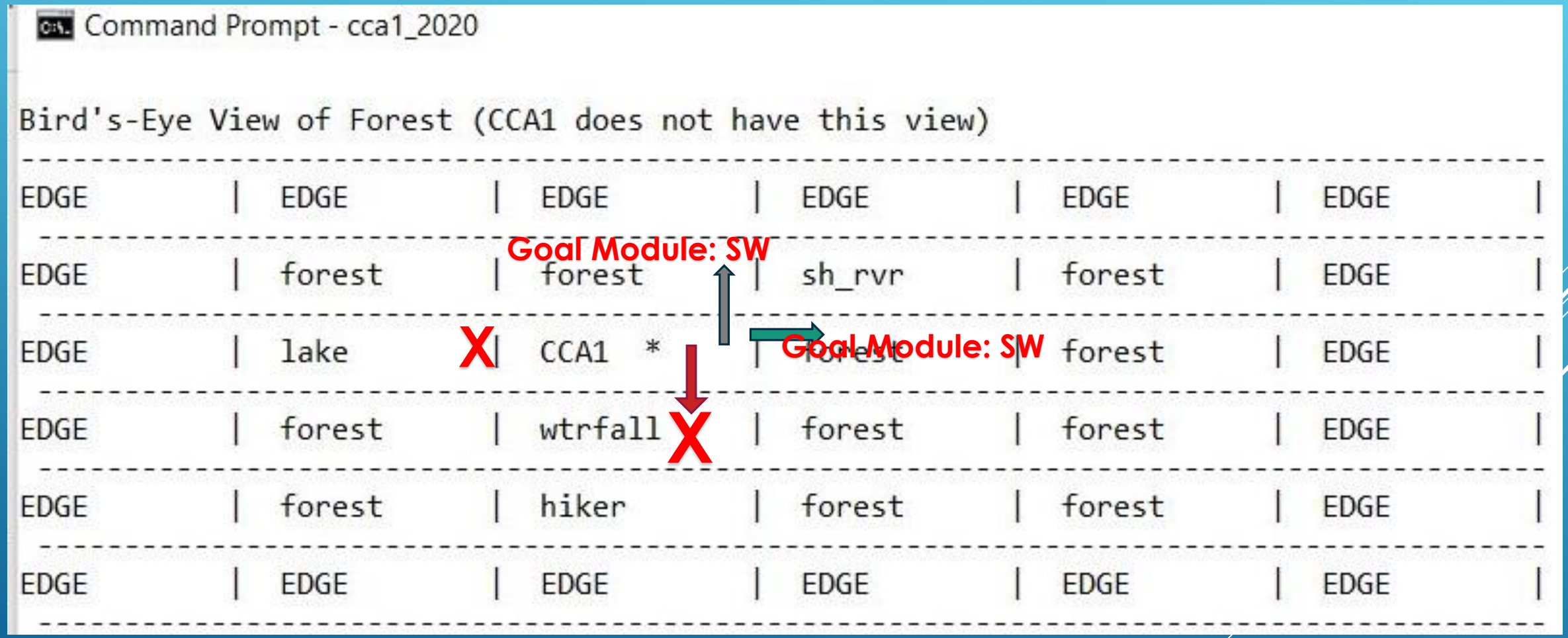


“do not go”

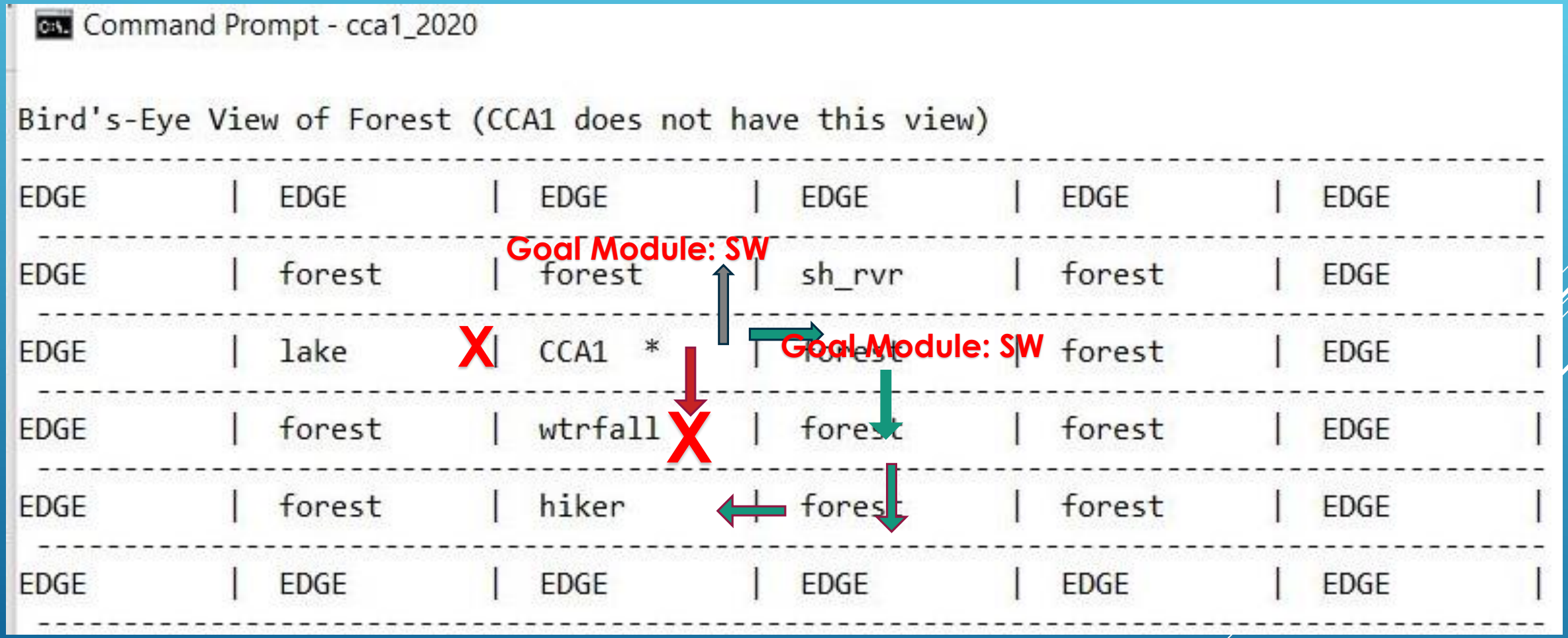
->retrieve previous temporary map

->do not go south

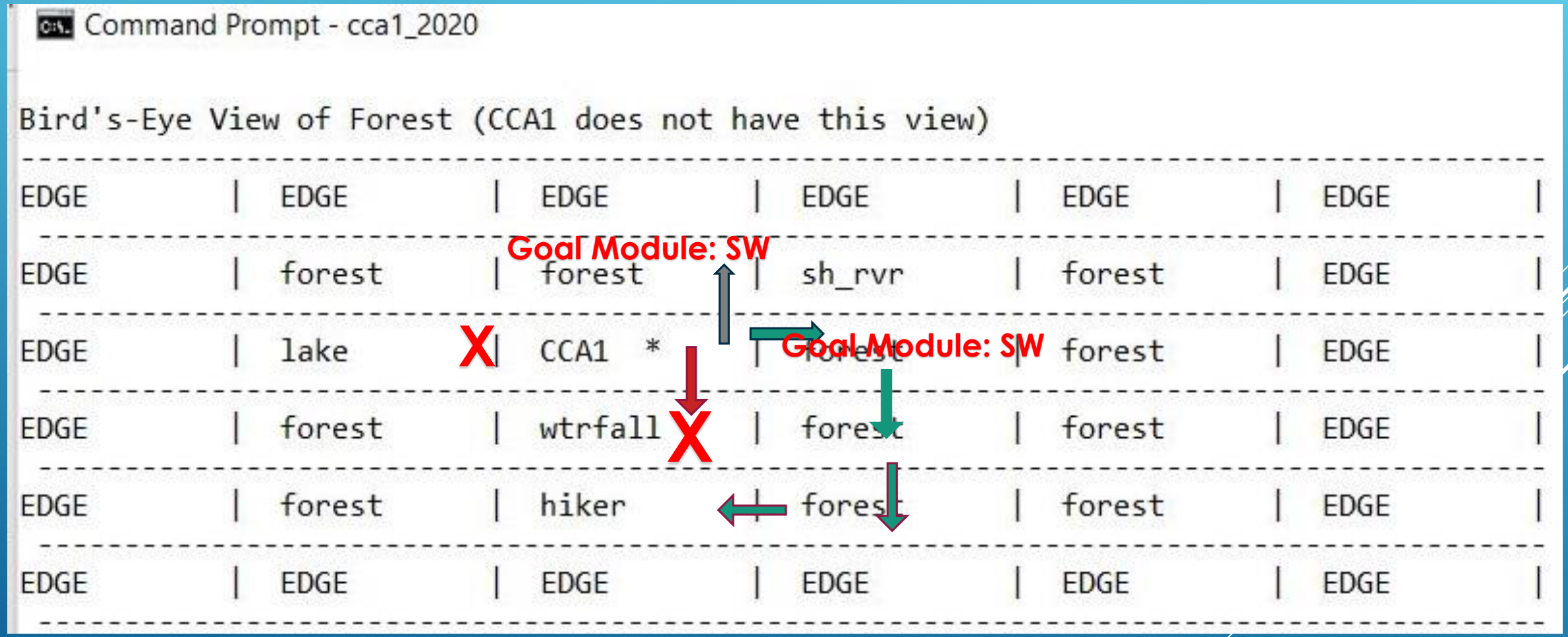
Do not go south – goes east even though bias from Goal Module to go south or west.



Continues south and then west…. and…. Rescues the lost hiker



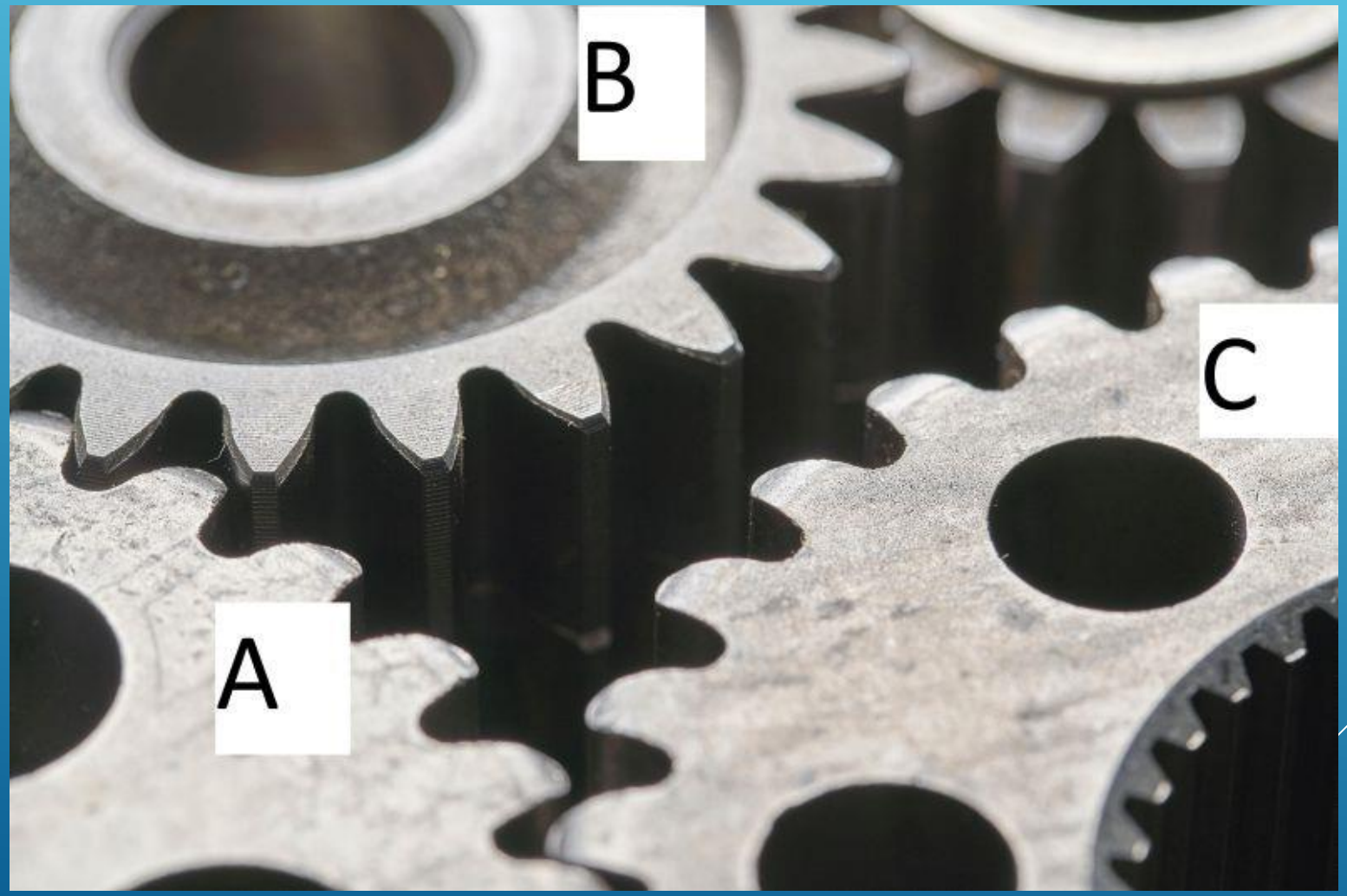
Even though CCA1 had never seen a waterfall before, **it causally avoided this danger**



-Causality emerges from the architecture of the CCA1

-No central controlling stored program other than the repeating processing cycles of the CCA1

New simulation – CCA1 is operating a machine repair robot.
Inspecting a broken machine it has never seen before.
If Gear C is turned, what happens to Gear B?

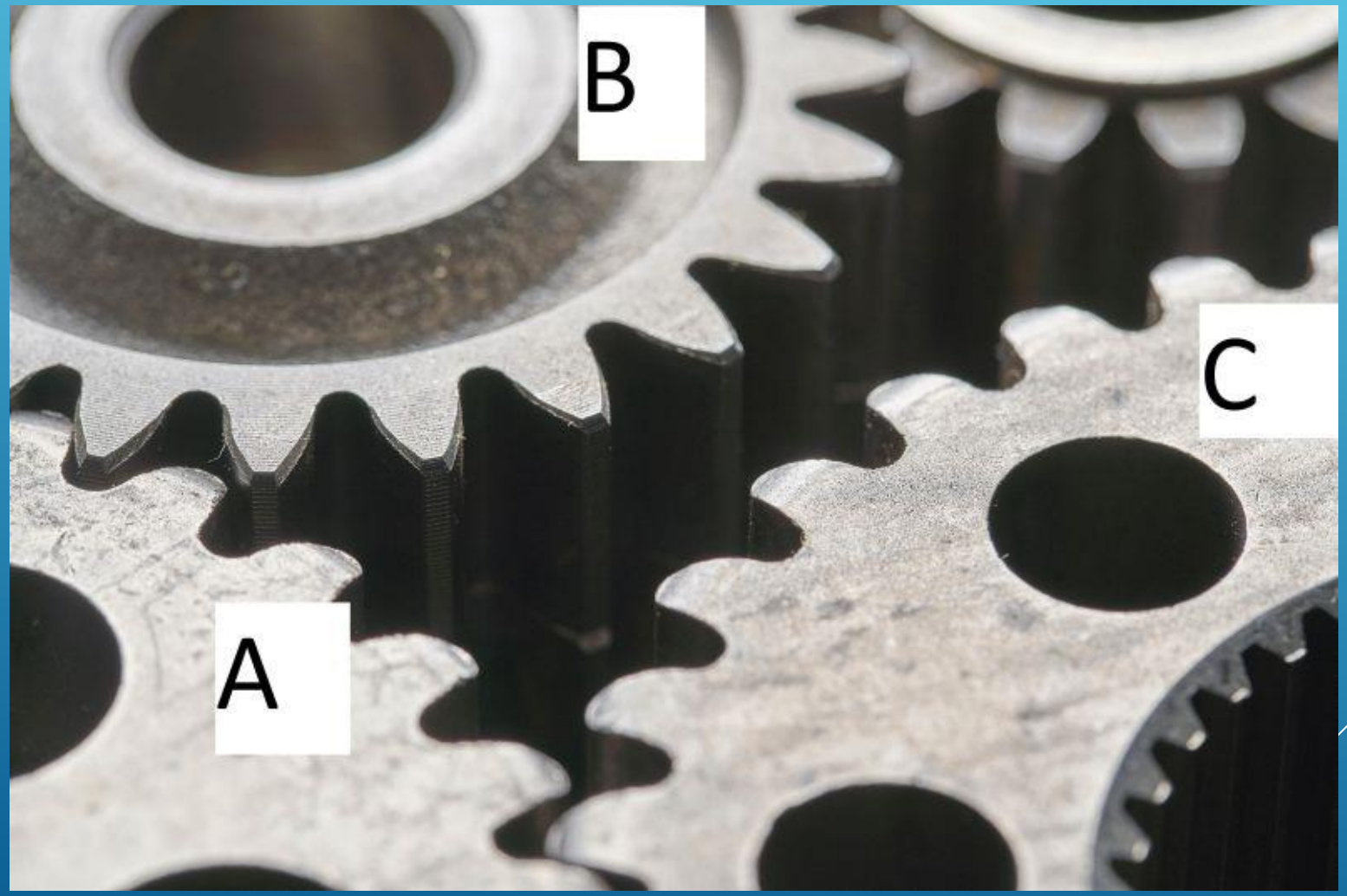


Small experiment – it pushes on (turns) Gear A and Gear B moves (turns)

Command Prompt - cca1_2020

```
Internal Map From Stack
-------------------------------------------------------------------------------
air*        |  air        |  air        |  air        |  air        |  air        |
 ------------------------------------------------------------------------------
air         |  air        |  air        |  air        |  air        |  air        |
 ------------------------------------------------------------------------------
air         |  A;moves    |  B;moves    |  air        |  air        |  air        |
 ------------------------------------------------------------------------------
air         |  air        |  air        |  air        |  air        |  air        |
 ------------------------------------------------------------------------------
air         |  air        |  hiker      |  air        |  air        |  air        |
 ------------------------------------------------------------------------------
air         |  air        |  air        |  air        |  air        |  air        |
 ------------------------------------------------------------------------------
air         |  air        |  air        |  air        |  air        |  air        |
```

New simulation – CCA1 is operating a machine repair robot.
Inspecting a broken machine it has never seen before.
If Gear C is turned, what happens to Gear B?



Gear C is recognized and added to create a new temporary map

Command Prompt - cca1_2020

```
Internal Map From Stack
-------------------------------------------------------------------------------
air*        |  air        |  air        |  air        |  air        |  air        |
 ------------------------------------------------------------------------------
air         |  *push      |  air        |  air        |  air        |  air        |
 ------------------------------------------------------------------------------
C           |  A;moves    |  B;moves    |  air        |  air        |  air        |
 ------------------------------------------------------------------------------
air         |  air        |  air        |  air        |  air        |  air        |
 ------------------------------------------------------------------------------
air         |  air        |  air        |  air        |  air        |  air        |
 ------------------------------------------------------------------------------
air         |  air        |  air        |  air        |  air        |  air        |
 ------------------------------------------------------------------------------
air         |  air        |  air        |  air        |  air        |  air        |
```

If push (turn) Gear C then Gear C moves (turns) and so will its specific neighbor move (turn)

Command Prompt - cca1_2020

Internal Map From Stack

air*	air	air	air	air	air
*push	air	air	air	air	air
C;moves	A;moves	air	air	air	air
air	air	air	air	air	air
air	air	air	air	air	air
air	air	air	air	air	air
air	air	air	air	air	air

Update with previous temporary map. New temporary map shows that if Gear C is moved (turned), then Gear B will move (turn)

Command Prompt - cca1_2020

```
Internal Map From Stack
-------------------------------------------------------------------------------
air*        |  air        |  air        |  air        |  air        |  air        |
 ------------------------------------------------------------------------------
*push       |  air        |  air        |  air        |  air        |  air        |
 ------------------------------------------------------------------------------
C;moves     |  A;moves    |  B;moves    |  air        |  air        |  air        |
 ------------------------------------------------------------------------------
air         |  air        |  air        |  air        |  air        |  air        |
 ------------------------------------------------------------------------------
air         |  air        |  air        |  air        |  air        |  air        |
 ------------------------------------------------------------------------------
air         |  air        |  air        |  air        |  air        |  air        |
 ------------------------------------------------------------------------------
air         |  air        |  air        |  air        |  air        |  air        |
 ------------------------------------------------------------------------------
```

-Cannot fully repair a machine with 100's of parts by associations only (unless very common reasons for the breakdowns)

– even if only move a few parts there are millions and millions of combinations that need to be tried and learned by association ->simply not possible/practical

Causality allows repairing a machine the CCA1 has never seen before.

Causality emerges from the architecture

CCA1 allows transition from Associative Behavior (using a subset of the architecture) to Higher Cognitive Level Behavior (using the full causal features of the architecture)

Consider the ability to make analogies

eg, The previous search and rescue CCA1 robot returns from its mission in the forest (as above) and then is asked if it should spend more of its free time with Person A or Person B.

The persons are similar but Person B is more smiley but also very noisy

This is more in the realm of philosophers or poets!!

-Object A (person A) and Object B (person B) put onto a temporary map in the Navigation Center

-CCA1 decides whether it should navigate to Object B:

-Instinctive Primitives like smiling people

-However, Object B is noisy

→thus pulls up the previous temporary map it had: the river seemed safe but made much noise also, and was considered a danger

→thus pulls up the previous temporary map it had: the river seemed safe but made much noise also, and was considered a danger

→pulls up previous map in stack → Object B is considered a danger

→fed back to sensory modules → in the next processing cycle processed and triggers in the Instinctive Primitives to stay away from Object B →Navigation Module decision to navigate to Object A → CCA1 answer to prefer Person A

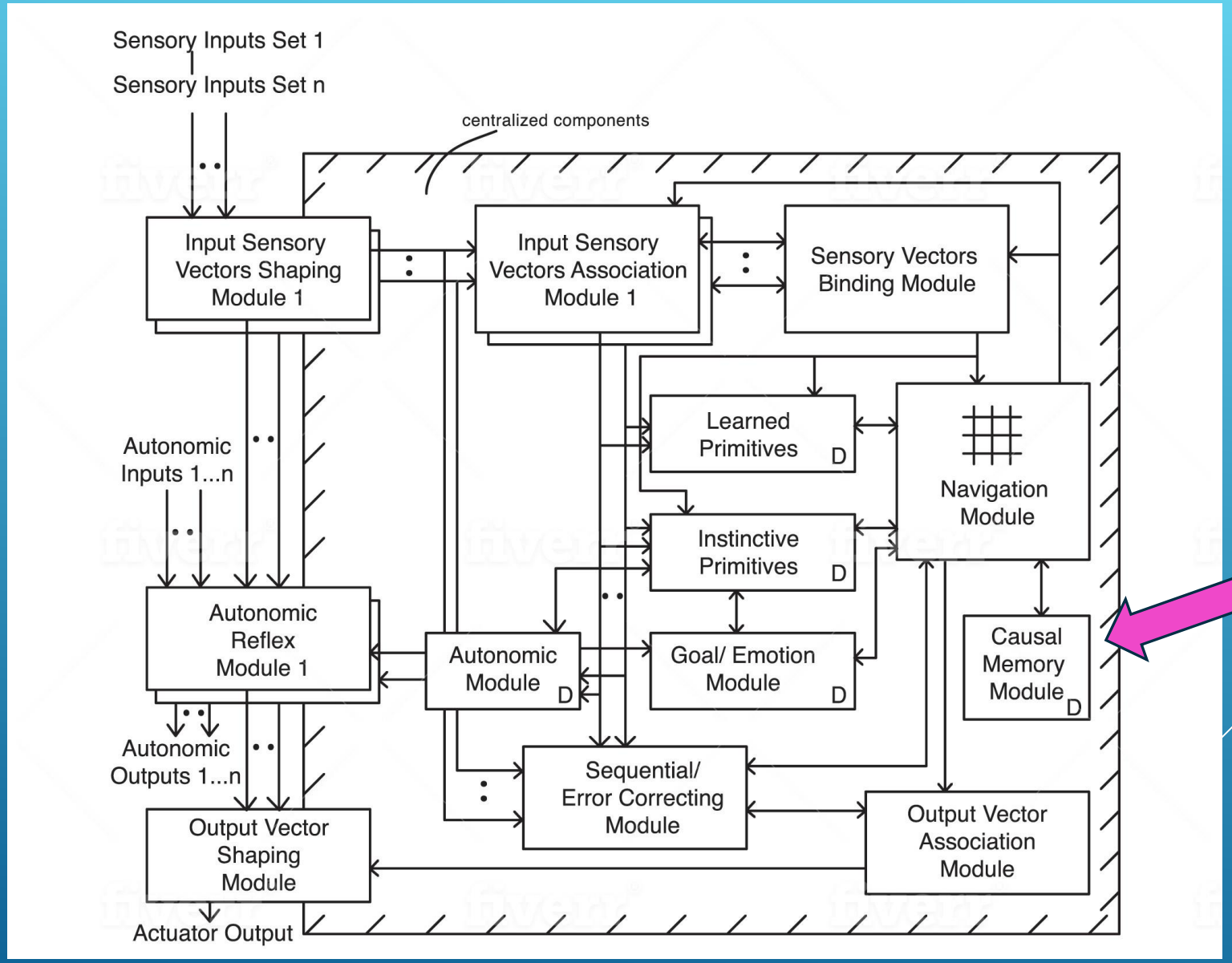
Analogies and other high level cognitive processes emerge directly from the architecture of the CCA1

Analogies emerged without any specialized algorithm or any special central controlling program, other than the relatively simple repeating cycles of the architecture

Explainability

After being used, ‘temporary maps’ are actually stored permanently in the Causal Memory portion of the Navigation Module

Causal Cognitive Architecture 1 (CCA1)



Useful to retrieve stored ‘temporary maps’ if there is a similar situation

Stored maps give an **excellent explanation** of why the CCA1 took a decision
(albeit, unable to explain non-causal sensory portion of the architecture)

CCA1 Supports Schneider Psychosis Hypothesis

Schneider –BICA 2019:
-Imperfect functioning in going from precausal to full causal behavior (more complexity, feeding partial results back to sensory modules) can result in psychotic behavior (hallucinations, delusions and reduced cognition)

CCA1 Supports Schneider Psychosis Hypothesis

-10% of humans at some point psychosis or psychosis-like symptoms
-0% of animals have psychosis (but have analogies of almost all other mental disorders)
-0% of animals capable of causal behavior

CCA1 Supports Schneider Psychosis Hypothesis

CCA1, while much more straightforward than Schneider 2019 MBCA – enough complexity to allow psychosis to emerge if any of a wide variety of small malfunctions in circuitry

Causal Cognitive Architecture 1 (CCA1)

-Allows causality to emerge from a system without any central controlling stored program (other than repeating sensory cycles)
-Tight integration of connectionist elements into a system capable of causal, symbolic operations

Causal Cognitive Architecture 1 (CCA1)

-High level cognitive processes emerge directly from the architecture
-Supports Schneider's psychosis hypothesis
-Provides a plausible pathway for the natural evolution of vertebrate brains
-Provides tool to explore pathways to AGI

CCA1 *closes* the Neural Symbolic Gap

· **Neural Network** – phenomenal image processing and reinforcement learning

· **Child** – phenomenal causal learning with few examples (eg, Gopnik)

Future Work

- Larger, more comprehensive simulation
- More formal proofs of the features of the CCA1

howard.schneider@gmail.com