Affra talks about abstraction, in the form of a basic “chunking” mechanism called the macro. Qiana goes along for the ride.

Qiana: OK. Now what?

Affra: Do you understand what I mean when I write \( \langle S2 \ P \ LP \ P \ RP \ X2 \rangle \Rightarrow \langle C.5 \ B.5 \ \rangle \)?

Qiana: Do you mean that the sequence of primitive Clay commands to the left of the double right arrow produce the sequence of sounds represented on the right of the arrow?

Affra: Yes!!

Qiana: Do you really need the \( \langle RP \ X2 \rangle \)?

Affra: If I want the state of the note to be the same after the sequence of commands is performed as it was before the sequence of commands was performed – then I do! If not, then I don’t.

Qiana: Hmmm ...

Affra: It is a concept called invariance. It is one of the most important concepts that there is in computer programming and problem solving!

Qiana: OK. I’ll take your word for it – you haven’t lied to me yet!

Affra: Just for the record: A program is invariant with respect to a property if the state of that property is unchanged as a result of running the program.

Qiana: Like closing a door after you walk throught a closed door! The state of the door (closed) is the same after the operation (you passing through the door) as it was before the operation.

Affra: Exactly!

Affra: Do you understand what I mean when I write \( \langle S2 \ P \ LP \ P \ RP \ RP \ P \ LP \ P \ RP \ LP \ X2 \rangle \Rightarrow \langle C.5 \ B.5 \ / \ D.5 \ / \ C.5 \ \rangle \)?

Qiana: Sure. The sequence of Clay commands produces the sequence of sounds.

Affra: And do you understand \( \langle S2 \ P \ LP \ P \ RP \ RP \ P \ LP \ P \ RP \ RP \ P \ LP \ P \ RP \ LP \ LP \ X2 \rangle \Rightarrow \langle C.5 \ B.5 \ / \ D.5 \ / \ C.5 \ / \ E.5 \ / \ D.5 \ \rangle \)?
Qiana: Yes.

Affra: And \( \langle \text{S2 P LP P RP RP P LP P RP P LP P RP LP LP X2 P} \rangle \Rightarrow \langle \text{C.5 \ B.5 / D.5 \ C.5 / E.5 \ D.5 \ C1} \rangle \)?

Qiana: Yes. But, to paraphrase Winnie the Pooh, “I am a Qiana of very little brain, and long sequences bother me!”.

Affra: Aha! Yes!! Indeed!!! I was hoping you would stop me – and I was hoping that you would do it at exactly this time.

Qiana: You were?

Affra: Yes. We are all like the famous bear of whom you speak, in many respects. What I mean to say is that humans (and bears???) are simply not particularly good at processing too much detail at any one time. Of relevance to our present conversation, humans tend to get bogged down trying to wrap their minds around long sequences of commands – even the most detail oriented humans.

Qiana: But don’t you need to write long sequences of commands to produce long melodic lines?

Affra: Actually, no. The “secret” is in abstraction!

Qiana: Oh no! I feel a bit of a lecture coming on!

Affra: I’ll try to spare you.

Qiana: No worries!

Affra: Sequences of primitive commands quickly become too cumbersome to work with effectively. The concept of the macro serves to remedy this situation.

Affra: A macro is a symbol which denotes a sequence of symbols.

Qiana: Hmmmm ...

Affra: Do you understand how I conceived of my sequence of 7 notes?

Qiana: Pray tell.

Affra: I imagined playing the sequence \( \langle \text{C / D / E \ C} \rangle \), but with each of the first three notes followed by its predecessor (within its scale) and with the last note followed by itself – except for the rhythmic aspect of the sequence.

Qiana: I see.
Affra: So I could have written the Clay code as \( \langle S2 \ DSC2 \ RP \ DSC2 \ RP \ DSC2 \ LP \ LP \ X2 \ P \rangle \) – provided I also were to write \( \langle DSC2 >> \ P \ LP \ P \ RP \rangle \).

Qiana: What is that last bit?

Affra: \( \langle DSC2 >> \ P \ LP \ P \ RP \rangle \) is an example of a macro definition.

Affra: In Clay, macro definition takes the form symbol >> sequence.

Affra: When Clay happens upon a macro, it simply replaces the macro symbol by the macro sequence, its “definition”.

Affra: Thus, Clay transforms \( \langle S2 \ DSC2 \ RP \ DSC2 \ RP \ DSC2 \ LP \ LP \ X2 \ P \rangle \) into \( \langle S2 \ P \ LP \ P \ RP \ RP \ P \ LP \ P \ RP \ P \ LP \ P \ RP \ LP \ LP \ X2 \ P \rangle \).

Qiana: I like it! The former sequence is certainly simpler to read than the latter.

Affra: It is useful to name things.

Qiana: Long pause. Perhaps we should name our sequence? Something like this: \( \langle SEQ >>> S2 \ DSC2 \ RP \ DSC2 \ RP \ DSC2 \ LP \ LP \ X2 \ P \rangle \)?

Affra: Absolutely!

Qiana: And then by typing \( \langle SEQ \rangle \) I will hear \( \langle C.5 \ \ B.5 / D.5 \ C.5 / E.5 \ D.5 \ C1 \rangle \)?

Affra: Yep! And now you can use \( \langle SEQ \rangle \) to generate an even longer sequence.

Qiana: Something like \( \langle SEQUENCE >>> S2 \ SEQ \ RP \ SEQ \ LP \ LP \ SEQ \ RP \ SEQ \ X2 \rangle \)?


Affra: You might like to try playing the sequence with different instruments in the context of a variety of alphabets. Perhaps \( \langle G\text{-MINOR VIOLIN SEQUENCE} \rangle \). Or \( \langle D\text{-WHOLETONE WHISTLE SEQUENCE} \rangle \). Or \( \langle F\text{-BLUES MUTEDGUITAR SEQUENCE} \rangle \). Or \( \langle E\text{-DIMTRIAD HAMMONDORGAN SEQUENCE} \rangle \). Or ...

Qiana: Cool!