

## Base Language - Feature #1597

Feature # 1596 (Closed): complete big decimal implementation

### shift SQRT implementation (MathOps.sqrt()) to BigDecimal

10/19/2012 08:11 AM - Greg Shah

<b>Status:</b>	Closed	<b>Start date:</b>	01/24/2014
<b>Priority:</b>	Normal	<b>Due date:</b>	01/28/2014
<b>Assignee:</b>	Hynek Cihlar	<b>% Done:</b>	100%
<b>Category:</b>		<b>Estimated time:</b>	6.00 hours
<b>Target version:</b>	Cleanup and Stabilization for Server Features		
<b>billable:</b>	No	<b>vendor_id:</b>	GCD
<b>Description</b>			

#### History

##### #1 - 10/31/2012 01:50 PM - Greg Shah

- Target version set to Milestone 7

##### #2 - 04/25/2013 10:59 AM - Greg Shah

- Target version changed from Milestone 7 to Milestone 11

##### #3 - 05/03/2013 10:04 AM - Greg Shah

- Assignee set to Ovidiu Maxiniuc

- Start date changed from 10/19/2012 to 08/06/2013

- Due date set to 08/06/2013

##### #4 - 01/21/2014 01:31 PM - Greg Shah

- Due date changed from 08/06/2013 to 01/28/2014

- Start date changed from 08/06/2013 to 01/24/2014

- Assignee changed from Ovidiu Maxiniuc to Hynek Cihlar

##### #5 - 01/24/2014 07:58 AM - Hynek Cihlar

What is the expected precision of the algorithm?

##### #6 - 01/24/2014 08:29 AM - Greg Shah

I don't know the answer. Often, if the input is a decimal value, then the output can have the same precision. Otherwise, usually the precision is the decimal default (10 digits).

Please write some test code to explore this and to prove that the result is compatible with the 4GL for a representative range of input values.

##### #7 - 01/24/2014 06:43 PM - Hynek Cihlar

- Status changed from New to WIP

##### #8 - 02/02/2014 06:42 PM - Hynek Cihlar

- File hc\_upd20140202a.zip added

Attached are the changes for initial review.

The implemented algorithm yields the same results as the native Progress code for numbers less than 9999999999999999 or so. For greater numbers, the implemented algorithm calculates results with smaller errors.

I attempted to modify the conversion to emit the sqrt call with a decimal argument (ex. `sqrt(new decimal("1"))`) but didn't succeed. See `SignatureHelper.java` in the attached file.

**#9 - 02/03/2014 05:06 AM - Hynek Cihlar**

- File `hc_upd20140203a.zip` added

I didn't realize the evaluator already casts the operands to proper types. Attached is the updated code.

**#10 - 02/03/2014 09:51 AM - Hynek Cihlar**

- File `hc_upd20140203b.zip` added

Attached file changes the precision of the sqrt algorithm to be defined by `decimal.MAX_SCALE`.

**#11 - 02/03/2014 04:45 PM - Hynek Cihlar**

- Status changed from *WIP* to *Review*

**#12 - 02/03/2014 06:46 PM - Greg Shah**

Code Review 0203b

Overall, this is very good.

1. Should we put a quick out for `BigDecimal.ONE`?
2. Add javadoc to describe the 4GL behavior of `SQRT` and how this algorithm matches (and more importantly, deviates). For example, for negative numbers the 4GL will return unknown value. This algorithm does the same (indirectly) by returning null, which will yield a decimal instance that is unknown.
3. Please check in your 4GL testcases.

**#13 - 02/05/2014 07:20 PM - Hynek Cihlar**

- File `hc_upd20140205a.zip` added

The input of one is covered by the initial approximation, but yes it won't hurt if stated explicitly.

New in the attached file:

- quick out for one
- more docs including comparison with native Progress
- quick out for unknown value because `sqrt(?) == 1`

Also the testcases file is checked in. See `testcases/uast/math/sqrt.p`.

**#14 - 02/06/2014 08:15 AM - Greg Shah**

Code Review 0205a

This looks good.

My only question is about this table:

* Progress	Error: 0.0000209975	0.0010625323	0.2415973135	28.5870600263	6385341029806439311343616
* 4GL	Error: 0.0000422480	0.0004300767	0.0026421128	0.00006997839	17801490614.8294175592

Should one of these lines show "P2J" instead of "Progress" or "4GL"? For us, "Progress" and "4GL" are the same thing and neither of them can be used to describe our P2J technology.

**#15 - 02/06/2014 10:02 AM - Hynek Cihlar**

- File *hc\_upd20140206a.zip* added

Yes, the "4GL" is a typo, thanks for catching it. I am attaching a fix and sending it to regression test.

**#16 - 02/06/2014 10:17 AM - Greg Shah**

Looks great!

**#17 - 02/12/2014 10:23 AM - Hynek Cihlar**

- File *hc\_upd20140209a.zip* added

The attached file is the final version.

Note, that the previously added condition of `sqrt(?) == 1` was removed. Progress behaves ok in this regard, just the test was incorrect.

The attached changes have been regression tested, once reviewed and approved it can be committed.

**#18 - 02/12/2014 10:37 AM - Greg Shah**

Code Review 0209a

The code looks good. Please commit and distribute it.

**#19 - 02/12/2014 02:04 PM - Hynek Cihlar**

Committed to revision 10461.

**#20 - 02/12/2014 02:06 PM - Hynek Cihlar**

- % Done changed from 0 to 100

**#21 - 02/12/2014 03:59 PM - Greg Shah**

- Status changed from Review to Closed

**#22 - 11/16/2016 12:07 PM - Greg Shah**

- Target version changed from Milestone 11 to Cleanup and Stabilization for Server Features

#### Files

---

hc_upd20140202a.zip	19.9 KB	02/02/2014	Hynek Cihlar
hc_upd20140203a.zip	9.98 KB	02/03/2014	Hynek Cihlar
hc_upd20140203b.zip	9.96 KB	02/03/2014	Hynek Cihlar
hc_upd20140205a.zip	10.4 KB	02/06/2014	Hynek Cihlar
hc_upd20140206a.zip	10.3 KB	02/06/2014	Hynek Cihlar
hc_upd20140209a.zip	68.2 KB	02/12/2014	Hynek Cihlar