Hej.

`<script>` λ `</script>`

JSHC

JavaScript Haskell Compiler

Vad är det?

En kompilator, som kompilerar

Haskell till JavaScript <script>

```
foo x = f x 1
```

```
foo = function(x) {
  return f(x)(1)
};
```

Vad är det?

En kompilator, som kompilerar

Haskell till JavaScript `<script>`

```
foo x = f x 1                    foo = function(x) {
                                   return f(x)(1)
                                 };
```

Skriven i JavaScript

Varför JavaScript?

Klientbaserade webbapplikationer

belastar servern mindre

Varför JavaScript?

JavaScript är de facto

standard för

klientbaserad skriptning

Varför JavaScript?

Den genererade koden
kan köras på klientsidan

Varför JavaScript?

Den genererade koden
kan köras på klientsidan

Koden kan genereras på klientsidan

Vad är JavaScript?

Vad är JavaScript?

Imperativt

Vad är JavaScript?

Imperativt

Objektorienterat

Vad är JavaScript?

Imperativt

Objektorienterat

Svagt dynamiskt typat

Vad är JavaScript?

Imperativt

Objektorienterat

Svagt dynamiskt typat

Första klassens funktioner

Ett exempel på JavaScript-kod ur verkliga livet: Autocomplete i ett textfält

```javascript
var suggestions = new Array("Boris", "Bäcker", "Peter", "Test","Bums");
var outp;
var oldins;
var posi = -1;
var words = new Array();
var input;
var key;
function setVisible(visi){
        var x = document.getElementById("shadow");
        var t = document.getElementsByName("text")[0];
        x.style.position = 'absolute';
        x.style.top =  (findPosY(t)+3)+"px";
        x.style.left = (findPosX(t)+2)+"px";
        x.style.visibility = visi;
}
function init(){
        outp = document.getElementById("output");
        window.setInterval("lookAt()", 100);
        setVisible("hidden");
        document.onkeydown = keygetter; //needed for Opera...
        document.onkeyup = keyHandler;
}

function findPosY(obj)
{
        var curtop = 0;
        if (obj.offsetParent){
                curtop += obj.offsetHeight;
                while (obj.offsetParent){
                        curtop += obj.offsetTop;
                        obj = obj.offsetParent;
                }
        }
        else if (obj.y){
                curtop += obj.y;
                curtop += obj.height;
        }
        return curtop;
}


        oldins = this.firstChild.nodeValue;
}
function lookAt(){
        var ins = document.getElementsByName("text")[0].value;
        if (oldins == ins) return;
        else if (posi > -1);
        else if (ins.length > 0){
                words = getWord(ins);
                if (words.length > 0){
                        clearOutput();
                        for (var i=0;i < words.length; ++i) addWord (words[i]);
                        setVisible("visible");
                        input = document.getElementsByName("text")[0].value;
                }
                else{
                        setVisible("hidden");
                        posi = -1;
                }
        }
        else{
                setVisible("hidden");
                posi = -1;
        }
        oldins = ins;
}
```

```javascript
function findPosX(obj)
{
        var curleft = 0;
        if (obj.offsetParent){
                while (obj.offsetParent){
                        curleft += obj.offsetLeft;
                        obj = obj.offsetParent;
                }
        }
        else if (obj.x)
                curleft += obj.x;
        return curleft;
}

function addWord(word){
        var sp = document.createElement("div");
        sp.appendChild(document.createTextNode(word));
        sp.onmouseover = mouseHandler;
        sp.onmouseout = mouseHandlerOut;
        sp.onclick = mouseClick;
        outp.appendChild(sp);
}

function clearOutput(){
        while (outp.hasChildNodes()){
                noten=outp.firstChild;
                outp.removeChild(noten);
        }
        posi = -1;
}

function getWord(beginning){
        var words = new Array();
        for (var i=0;i < suggestions.length; ++i){
                var j = -1;
                var correct = 1;
                while (correct == 1 && ++j < beginning.length){
                        if (suggestions[i].charAt(j) != beginning.charAt(j)) correct = 0;
                }
                if (correct == 1) words[words.length] = suggestions[i];
        }
        return words;
}


function setColor (_posi, _color, _forg){
        outp.childNodes[_posi].style.background = _color;
        outp.childNodes[_posi].style.color = _forg;
}

function keygetter(event){
        if (!event && window.event) event = window.event;
        if (event) key = event.keyCode;
        else key = event.which;
}
```

```javascript
function keyHandler(event){
        if (document.getElementById("shadow").style.visibility == "visible"){
                var textfield = document.getElementsByName("text")[0];
                if (key == 40){ //Key down
                        //alert (words);
                        if (words.length > 0 && posi  words.length-1){
                                if (posi >=0) setColor(posi, "#fff", "black");
                                else input = textfield.value;
                                setColor(++posi, "blue", "white");
                                textfield.value = outp.childNodes[posi].firstChild.nodeValue;
                        }
                }
                else if (key == 38){ //Key up
                        if (words.length > 0 && posi >= 0){
                                if (posi >=1){
                                        setColor(posi, "#fff", "black");
                                        setColor(--posi, "blue", "white");
                                        textfield.value = outp.childNodes[posi].firstChild.nodeValue;
                                }
                                else{
                                        setColor(posi, "#fff", "black");
                                        textfield.value = input;
                                        textfield.focus();
                                        posi--;
                                }
                        }
                }
                else if (key == 27){ // Esc
                        textfield.value = input;
                        setVisible("hidden");
                        posi = -1;
                        oldins = input;
                }
                else if (key == 8){ // Backspace
                        posi = -1;
                        oldins=-1;
                }
        }
}

var mouseHandler=function(){
        for (var i=0; i < words.length; ++i)
                setColor (i, "white", "black");

        this.style.background = "blue";
        this.style.color= "white";
}

var mouseHandlerOut=function(){
        this.style.background = "white";
        this.style.color= "black";
}

var mouseClick=function(){
        document.getElementsByName("text")[0].value = this.firstChild.nodeValue;
        setVisible("hidden");
        Posi = -1;
        oldins = this.firstChild.nodeValue;
}
```

2

```javascript
var suggestions = new Array("Boris", "Bäcker", "Peter", "Test","Bums");
var outp;
var oldins;
var posi = -1;
var words = new Array();
var input;
var key;
function setVisible(visi){
        var x = document.getElementById("shadow");
        var t = document.getElementsByName("text")[0];
        x.style.position = 'absolute';
        x.style.top =  (findPosY(t)+3)+"px";
        x.style.left = (findPosX(t)+2)+"px";
        x.style.visibility = visi;
}
function init(){
        outp = document.getElementById("output");
        window.setInterval("lookAt()", 100);
        setVisible("hidden");
        document.onkeydown = keygetter; //needed for Opera...
        document.onkeyup = keyHandler;
}

function findPosY(obj)
{
        var curtop = 0;
        if (obj.offsetParent){
                curtop += obj.offsetHeight;
                while (obj.offsetParent){
                        curtop += obj.offsetTop;
                        obj = obj.offsetParent;
                }
        }
        else if (obj.y){
                curtop += obj.y;
                curtop += obj.height;
        }
        return curtop;
}


        oldins = this.firstChild.nodeValue;
}
function lookAt(){
        var ins = document.getElementsByName("text")[0].value;
        if (oldins == ins) return;
        else if (posi > -1);
        else if (ins.length > 0){
                words = getWord(ins);
                if (words.length > 0){
                        clearOutput();
                        for (var i=0;i < words.length; ++i) addWord (words[i]);
                        setVisible("visible");
                        input = document.getElementsByName("text")[0].value;
                }
                else{
                        setVisible("hidden");
                        posi = -1;
                }
        }
        else{
                setVisible("hidden");
                posi = -1;
        }
        oldins = ins;
}
```

```javascript
function findPosX(obj)
{
        var curleft = 0;
        if (obj.offsetParent){
                while (obj.offsetParent){
                        curleft += obj.offsetLeft;
                        obj = obj.offsetParent;
                }
        }
        else if (obj.x)
                curleft += obj.x;
        return curleft;
}

function addWord(word){
        var sp = document.createElement("div");
        sp.appendChild(document.createTextNode(word));
        sp.onmouseover = mouseHandler;
        sp.onmouseout = mouseHandlerOut;
        sp.onclick = mouseClick;
        outp.appendChild(sp);
}

function clearOutput(){
        while (outp.hasChildNodes()){
                noten=outp.firstChild;
                outp.removeChild(noten);
        }
        posi = -1;
}

function getWord(beginning){
        var words = new Array();
        for (var i=0;i < suggestions.length; ++i){
                var j = -1;
                var correct = 1;
                while (correct == 1 && ++j < beginning.length){
                        if (suggestions[i].charAt(j) != beginning.charAt(j)) correct = 0;
                }
                if (correct == 1) words[words.length] = suggestions[i];
        }
        return words;
}


function setColor (_posi, _color, _forg){
        outp.childNodes[_posi].style.background = _color;
        outp.childNodes[_posi].style.color = _forg;
}

function keygetter(event){
        if (!event && window.event) event = window.event;
        if (event) key = event.keyCode;
        else key = event.which;
}
```

```javascript
function keyHandler(event){
        if (document.getElementById("shadow").style.visibility == "visible"){
                var textfield = document.getElementsByName("text")[0];
                if (key == 40){ //Key down
                        //alert (words);
                        if (words.length > 0 && posi  words.length-1){
                                if (posi >=0) setColor(posi, "#fff", "black");
                                else input = textfield.value;
                                setColor(++posi, "blue", "white");
                                textfield.value = outp.childNodes[posi].firstChild.nodeValue;
                        }
                }
                else if (key == 38){ //Key up
                        if (words.length > 0 && posi >= 0){
                                if (posi >=1){
                                        setColor(posi, "#fff", "black");
                                        setColor(--posi, "blue", "white");
                                        textfield.value = outp.childNodes[posi].firstChild.nodeValue;
                                }
                                else{
                                        setColor(posi, "#fff", "black");
                                        textfield.value = input;
                                        textfield.focus();
                                        posi--;
                                }
                        }
                }
                else if (key == 27){ // Esc
                        textfield.value = input;
                        setVisible("hidden");
                        posi = -1;
                        oldins = input;
                }
                else if (key == 8){ // Backspace
                        posi = -1;
                        oldins=-1;
                }
        }
}

var mouseHandler=function(){
        for (var i=0; i < words.length; ++i)
                setColor (i, "white", "black");

        this.style.background = "blue";
        this.style.color= "white";
}

var mouseHandlerOut=function(){
        this.style.background = "white";
        this.style.color= "black";
}

var mouseClick=function(){
        document.getElementsByName("text")[0].value = this.firstChild.nodeValue;
        setVisible("hidden");
        Posi = -1;
        oldins = this.firstChild.nodeValue;
}
```

Hitta typfelet!

```javascript
var suggestions = new Array("Boris", "Bäcker", "Peter", "Test","Bums");
var outp;
var oldins;
var posi = -1;
var words = new Array();
var input;
var key;
function setVisible(visi){
        var x = document.getElementById("shadow");
        var t = document.getElementsByName("text")[0];
        x.style.position = 'absolute';
        x.style.top =   (findPosY(t)+3)+"px";
        x.style.left = (findPosX(t)+2)+"px";
        x.style.visibility = visi;
}
function init(){
        outp = document.getElementById("output");
        window.setInterval("lookAt()", 100);
        setVisible("hidden");
        document.onkeydown = keygetter; //needed for Opera...
        document.onkeyup = keyHandler;
}

function findPosY(obj)
{
        var curtop = 0;
        if (obj.offsetParent){
                curtop += obj.offsetHeight;
                while (obj.offsetParent){
                        curtop += obj.offsetTop;
                        obj = obj.offsetParent;
                }
        }
        else if (obj.y){
                curtop += obj.y;
                curtop += obj.height;
        }
        return curtop;
}


        oldins = this.firstChild.nodeValue;
}
function lookAt(){
        var ins = document.getElementsByName("text")[0].value;
        if (oldins == ins) return;
        else if (posi > -1);
        else if (ins.length > 0){
                words = getWord(ins);
                if (words.length > 0){
                        clearOutput();
                        for (var i=0;i < words.length; ++i) addWord (words[i]);
                        setVisible("visible");
                        input = document.getElementsByName("text")[0].value;
                }
                else{
                        setVisible("hidden");
                        posi = -1;
                }
        }
        else{
                setVisible("hidden");
                posi = -1;
        }
        oldins = ins;
}
```

```javascript
function findPosX(obj)
{
        var curleft = 0;
        if (obj.offsetParent){
                while (obj.offsetParent){
                        curleft += obj.offsetLeft;
                        obj = obj.offsetParent;
                }
        }
        else if (obj.x)
                curleft += obj.x;
        return curleft;
}

function addWord(word){
        var sp = document.createElement("div");
        sp.appendChild(document.createTextNode(word));
        sp.onmouseover = mouseHandler;
        sp.onmouseout = mouseHandlerOut;
        sp.onclick = mouseClick;
        outp.appendChild(sp);
}

function clearOutput(){
        while (outp.hasChildNodes()){
                noten=outp.firstChild;
                outp.removeChild(noten);
        }
        posi = -1;
}

function getWord(beginning){
        var words = new Array();
        for (var i=0;i < suggestions.length; ++i){
                var j = -1;
                var correct = "1";
                while (correct == 1 && ++j < beginning.length){
                        if (suggestions[i].charAt(j) != beginning.charAt(j)) correct = 0;
                }
                if (correct == 1) words[words.length] = suggestions[i];
        }
        return words;
}


function setColor (_posi, _color, _forg){
        outp.childNodes[_posi].style.background = _color;
        outp.childNodes[_posi].style.color = _forg;
}

function keygetter(event){
        if (!event && window.event) event = window.event;
        if (event) key = event.keyCode;
        else key = event.which;
}
```

```javascript
function keyHandler(event){
        if (document.getElementById("shadow").style.visibility == "visible"){
                var textfield = document.getElementsByName("text")[0];
                if (key == 40){ //Key down
                        //alert (words);
                        if (words.length > 0 && posi  words.length-1){
                                if (posi >=0) setColor(posi, "#fff", "black");
                                else input = textfield.value;
                                setColor(++posi, "blue", "white");
                                textfield.value = outp.childNodes[posi].firstChild.nodeValue;
                        }
                }
                else if (key == 38){ //Key up
                        if (words.length > 0 && posi >= 0){
                                if (posi >=1){
                                        setColor(posi, "#fff", "black");
                                        setColor(--posi, "blue", "white");
                                        textfield.value = outp.childNodes[posi].firstChild.nodeValue;
                                }
                                else{
                                        setColor(posi, "#fff", "black");
                                        textfield.value = input;
                                        textfield.focus();
                                        posi--;
                                }
                        }
                }
                else if (key == 27){ // Esc
                        textfield.value = input;
                        setVisible("hidden");
                        posi = -1;
                        oldins = input;
                }
                else if (key == 8){ // Backspace
                        posi = -1;
                        oldins=-1;
                }
        }
}

var mouseHandler=function(){
        for (var i=0; i < words.length; ++i)
                setColor (i, "white", "black");

        this.style.background = "blue";
        this.style.color= "white";
}

var mouseHandlerOut=function(){
        this.style.background = "white";
        this.style.color= "black";
}

var mouseClick=function(){
        document.getElementsByName("text")[0].value = this.firstChild.nodeValue;
        setVisible("hidden");
        Posi = -1;
        oldins = this.firstChild.nodeValue;
}
```

Klipp och klistra

```javascript
var suggestions = new Array("Boris", "Bäcker", "Peter", "Test","Bums");
var outp;
var oldins;
var posi = -1;
var words = new Array();
var input;
var key;
function setVisible(visi){
        var x = document.getElementById("shadow");
        var t = document.getElementsByName("text")[0];
        x.style.position = 'absolute';
        x.style.top =  (findPosY(t)+3)+"px";
        x.style.left = (findPosX(t)+2)+"px";
        x.style.visibility = visi;
}
function init(){
        outp = document.getElementById("output");
        window.setInterval("lookAt()", 100);
        setVisible("hidden");
        document.onkeydown = keygetter; //needed for Opera...
        document.onkeyup = keyHandler;
}

function findPosY(obj)
{
        var curtop = 0;
        if (obj.offsetParent){
                curtop += obj.offsetHeight;
                while (obj.offsetParent){
                        curtop += obj.offsetTop;
                        obj = obj.offsetParent;
                }
        }
        else if (obj.y){
                curtop += obj.y;
                curtop += obj.height;
        }
        return curtop;
}


        oldins = this.firstChild.nodeValue;
}
function lookAt(){
        var ins = document.getElementsByName("text")[0].value;
        if (oldins == ins) return;
        else if (posi > -1);
        else if (ins.length > 0){
                words = getWord(ins);
                if (words.length > 0){
                        clearOutput();
                        for (var i=0;i < words.length; ++i) addWord (words[i]);
                        setVisible("visible");
                        input = document.getElementsByName("text")[0].value;
                }
                else{
                        setVisible("hidden");
                        posi = -1;
                }
        }
        else{
                setVisible("hidden");
                posi = -1;
        }
        oldins = ins;
}
```

```javascript
function findPosX(obj)
{
        var curleft = 0;
        if (obj.offsetParent){
                while (obj.offsetParent){
                        curleft += obj.offsetLeft;
                        obj = obj.offsetParent;
                }
        }
        else if (obj.x)
                curleft += obj.x;
        return curleft;
}

function addWord(word){
        var sp = document.createElement("div");
        sp.appendChild(document.createTextNode(word));
        sp.onmouseover = mouseHandler;
        sp.onmouseout = mouseHandlerOut;
        sp.onclick = mouseClick;
        outp.appendChild(sp);
}

function clearOutput(){
        while (outp.hasChildNodes()){
                noten=outp.firstChild;
                outp.removeChild(noten);
        }
        posi = -1;
}

function getWord(beginning){
        var words = new Array();
        for (var i=0;i < suggestions.length; ++i){
                var j = -1;
                var correct = 1;
                while (correct == 1 && ++j < beginning.length){
                        if (suggestions[i].charAt(j) != beginning.charAt(j)) correct = 0;
                }
                if (correct == 1) words[words.length] = suggestions[i];
        }
        return words;
}


function setColor (_posi, _color, _forg){
        outp.childNodes[_posi].style.background = _color;
        outp.childNodes[_posi].style.color = _forg;
}

function keygetter(event){
        if (!event && window.event) event = window.event;
        if (event) key = event.keyCode;
        else key = event.which;
}
```

```javascript
function keyHandler(event){
        if (document.getElementById("shadow").style.visibility == "visible"){
                var textfield = document.getElementsByName("text")[0];
                if (key == 40){ //Key down
                        //alert (words);
                        if (words.length > 0 && posi  words.length-1){
                                if (posi >=0) setColor(posi, "#fff", "black");
                                else input = textfield.value;
                                setColor(++posi, "blue", "white");
                                textfield.value = outp.childNodes[posi].firstChild.nodeValue;
                        }
                }
                else if (key == 38){ //Key up
                        if (words.length > 0 && posi >= 0){
                                if (posi >=1){
                                        setColor(posi, "#fff", "black");
                                        setColor(--posi, "blue", "white");
                                        textfield.value = outp.childNodes[posi].firstChild.nodeValue;
                                }
                                else{
                                        setColor(posi, "#fff", "black");
                                        textfield.value = input;
                                        textfield.focus();
                                        posi--;
                                }
                        }
                }
                else if (key == 27){ // Esc
                        textfield.value = input;
                        setVisible("hidden");
                        posi = -1;
                        oldins = input;
                }
                else if (key == 8){ // Backspace
                        posi = -1;
                        oldins=-1;
                }
        }
}

var mouseHandler=function(){
        for (var i=0; i < words.length; ++i)
                setColor (i, "white", "black");

        this.style.background = "blue";
        this.style.color= "white";
}

var mouseHandlerOut=function(){
        this.style.background = "white";
        this.style.color= "black";
}


var mouseClick=function(){
        document.getElementsByName("text")[0].value = this.firstChild.nodeValue;
        setVisible("hidden");
        Posi = -1;
        oldins = this.firstChild.nodeValue;
}
```

En lösning: Bibliotek

En lösning: Bibliotek

Abstraktion genom funktioner

En lösning: Bibliotek

Abstraktion genom funktioner

Osäkert gränssnitt

En annan lösning:

Översätt!

Varför just Haskell?

Varför just Haskell?

- Vi är Haskellprogrammerare!
  - Använda redan existerande kod

Varför just Haskell?

- Vi är Haskellprogrammerare!
  - Använda redan existerande kod

- Stark statisk typning

Varför just Haskell?

- Vi är Haskellprogrammerare!
  - Använda redan existerande kod

- Stark statisk typning

- Modulsystem

Varför just Haskell?

- Vi är Haskellprogrammerare!
  - Använda redan existerande kod

- Stark statisk typning

- Modulsystem

- Lat evaluering
  ```
  ones = 1:ones
  ```

Möjligheter att använda Haskell för webprogrammering finns redan

Möjligheter att använda Haskell för webprogrammering finns redan

Serverramverk
- Happstack
- Yesod

Möjligheter att använda Haskell för webprogrammering finns redan

Serverramverk
- Happstack
- Yesod

Kompilatorer till JavaScript
- GHCjs
- UHC

Nu är det dags för demon!

Kompilering: en översikt

Kompilering: en översikt

Tre steg

Kompilering: en översikt

Tre steg
- Abstrakt representation

Kompilering: en översikt

Tre steg
- Abstrakt representation

- Kontroll av representation

Kompilering: en översikt

Tre steg
- Abstrakt representation

- Kontroll av representation

- Generering av målrepresentation

Hur fungerar JSHC?
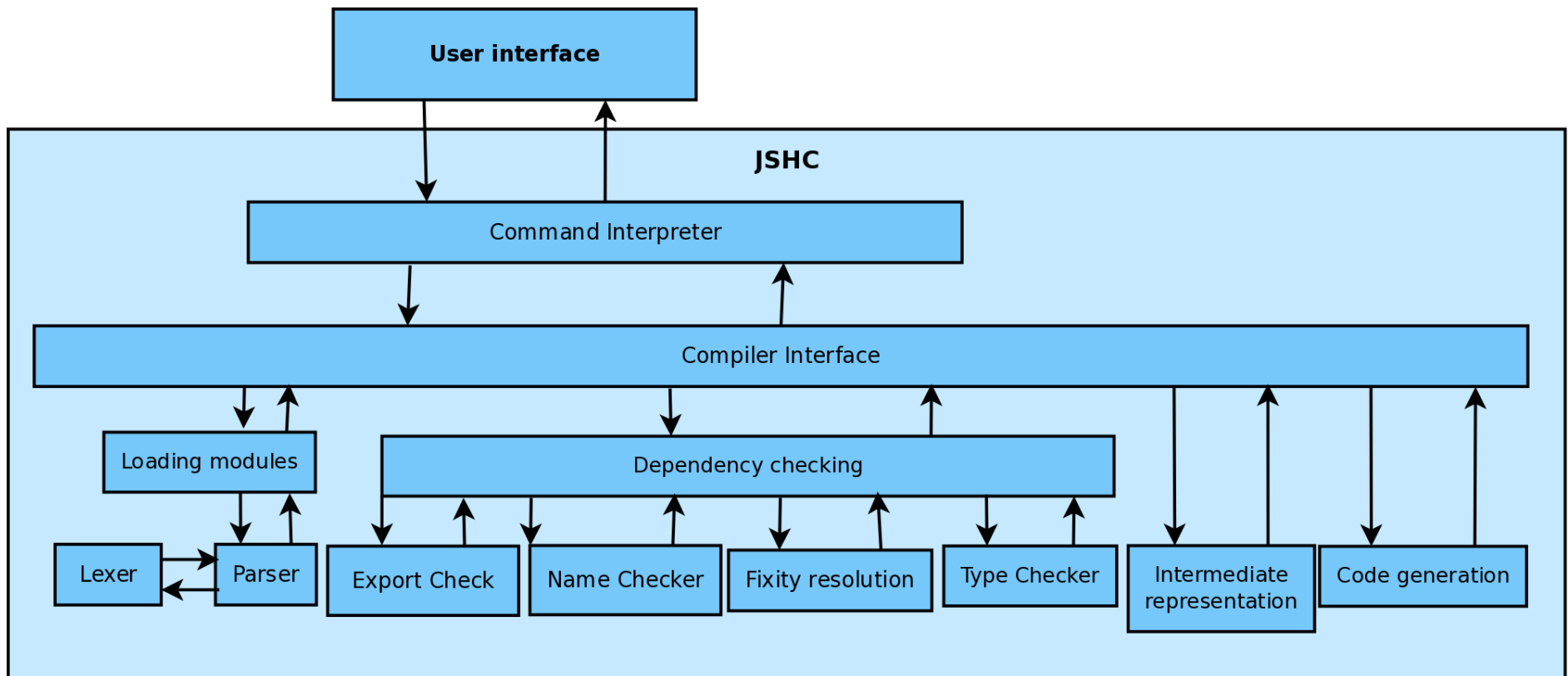
Hur fungerar JSHC?

Kommando skickas
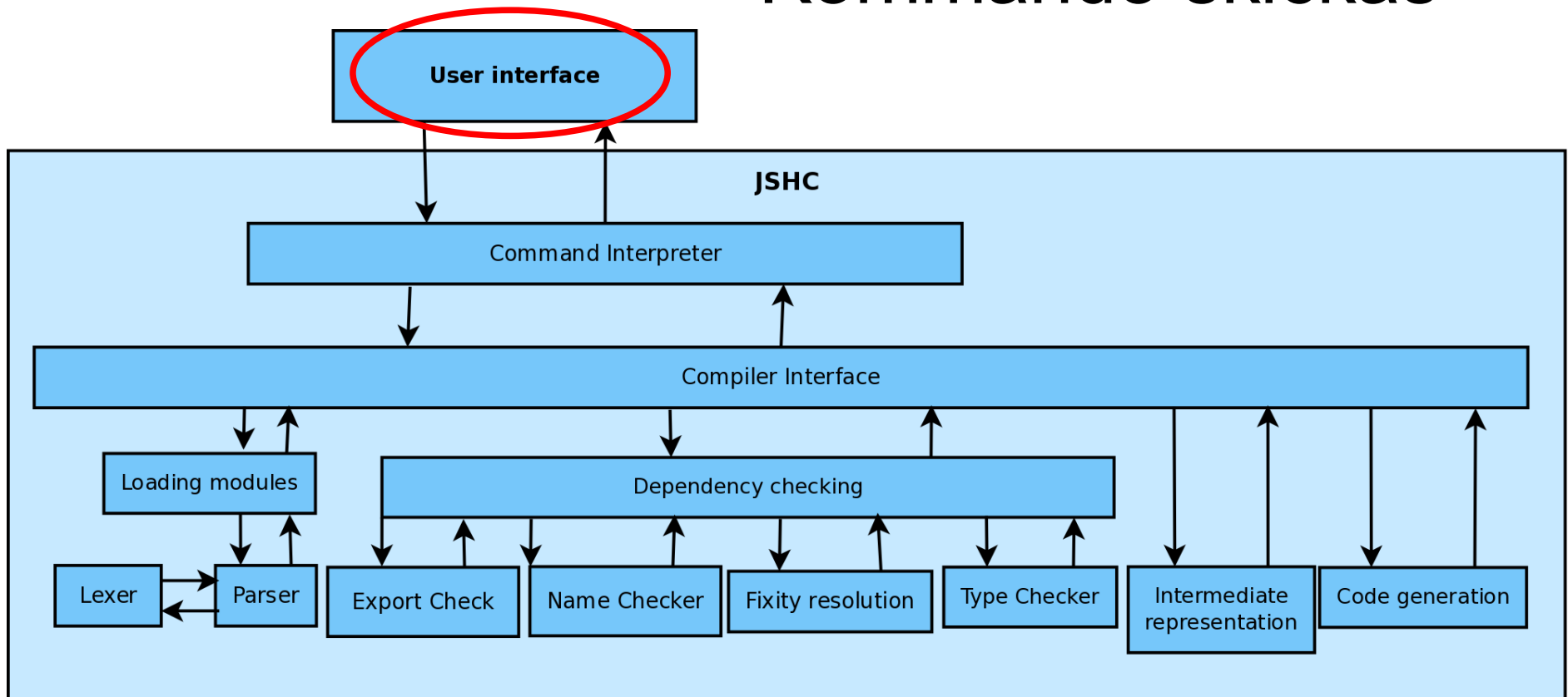
Hur fungerar JSHC?

Kommando tolkas

Hur fungerar JSHC?

Moduler laddas
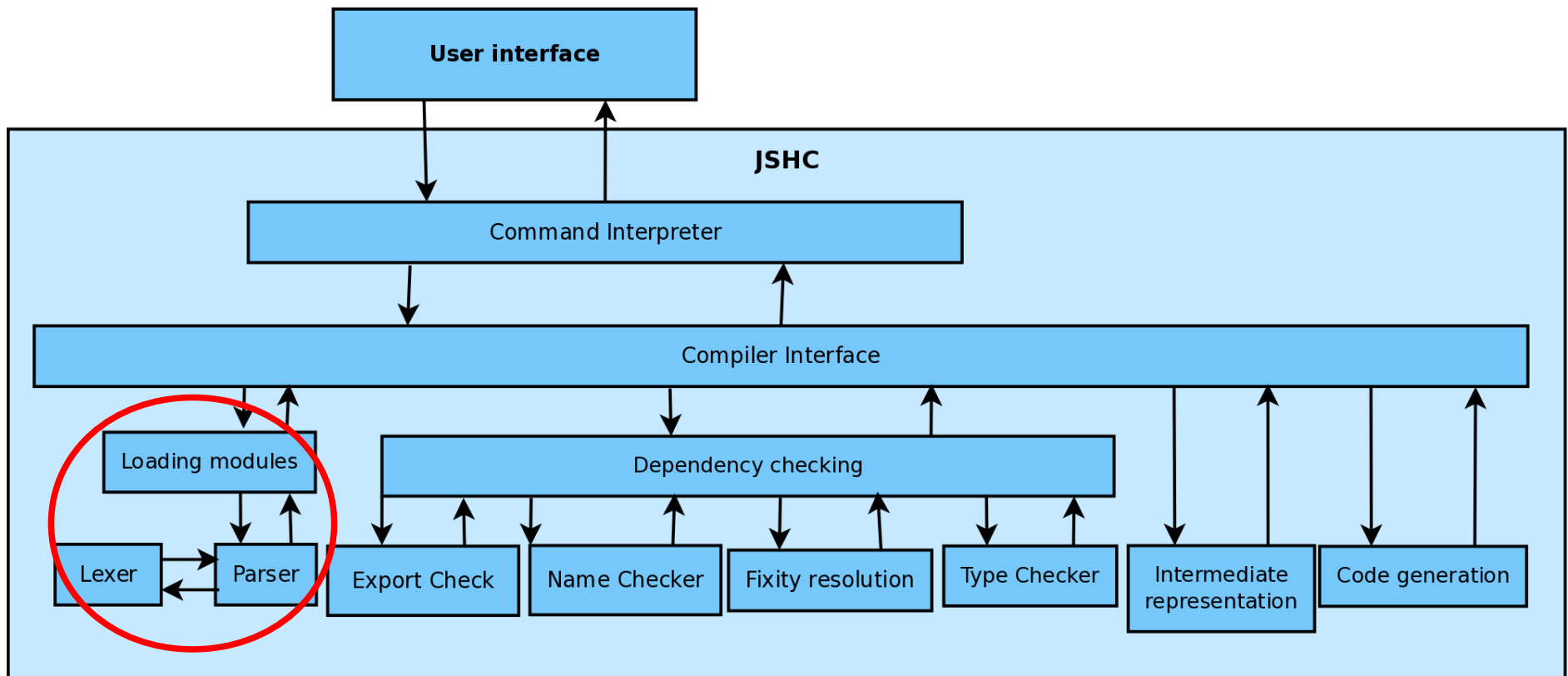
Hur fungerar JSHC?

Dependency checking
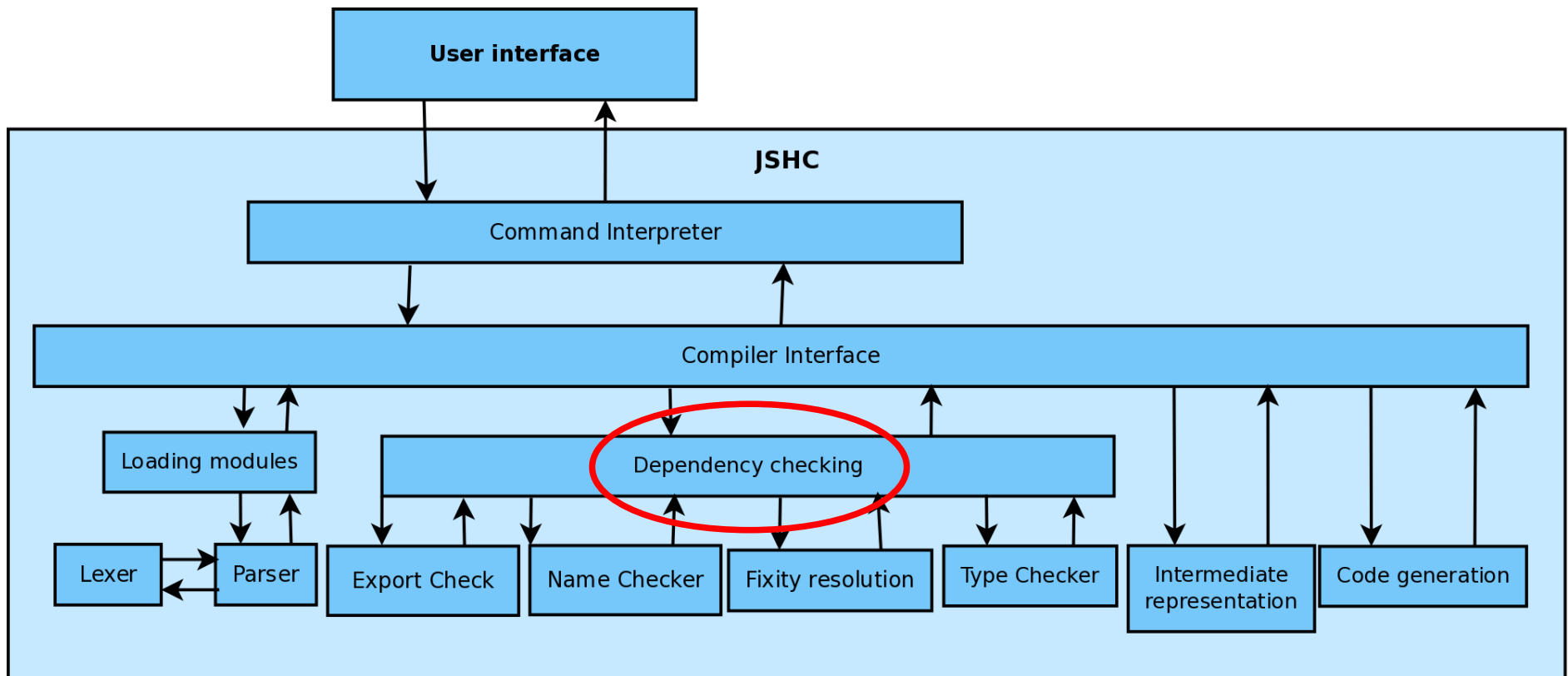
Dependency checking
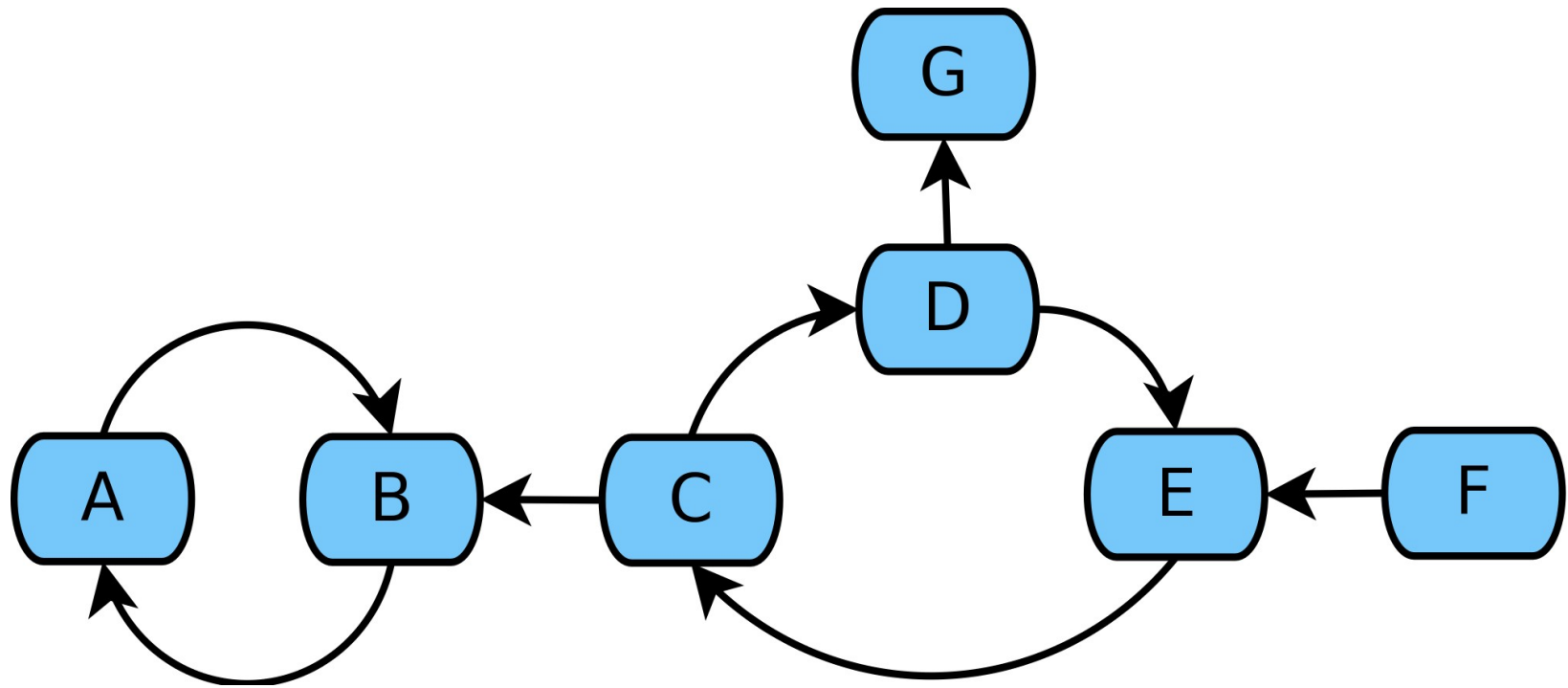
Hur fungerar JSHC?

Kontroll av exporter
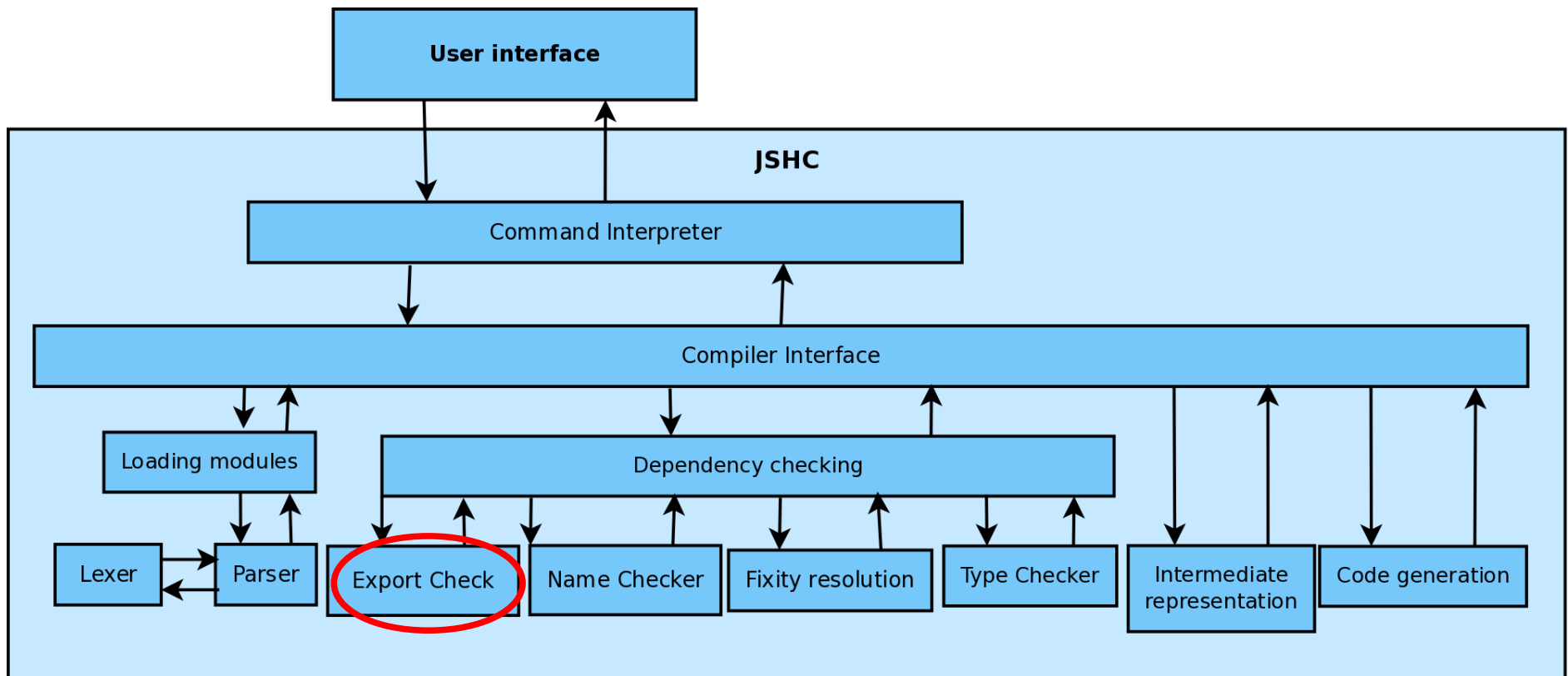
Hur fungerar JSHC?

Namnkontroll
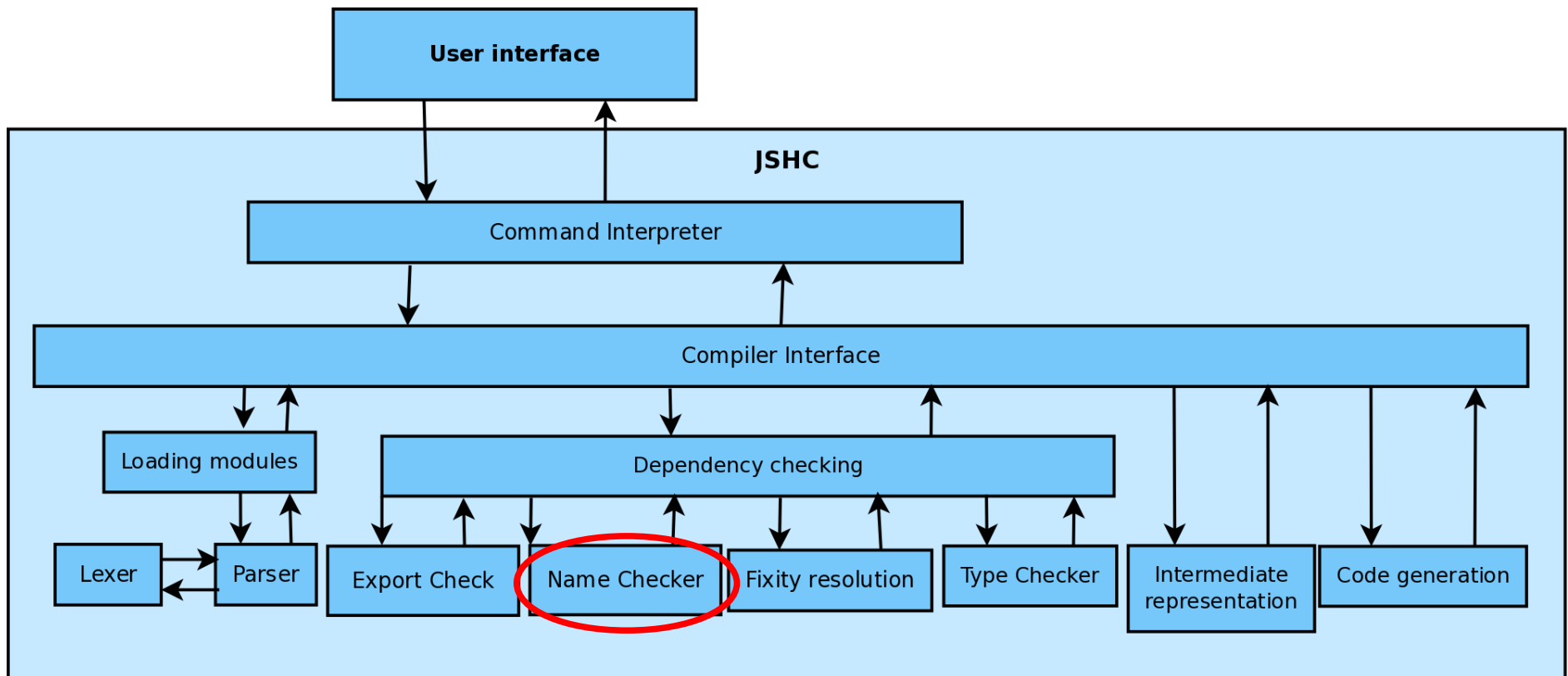
Hur fungerar JSHC?

Operatorprecedens
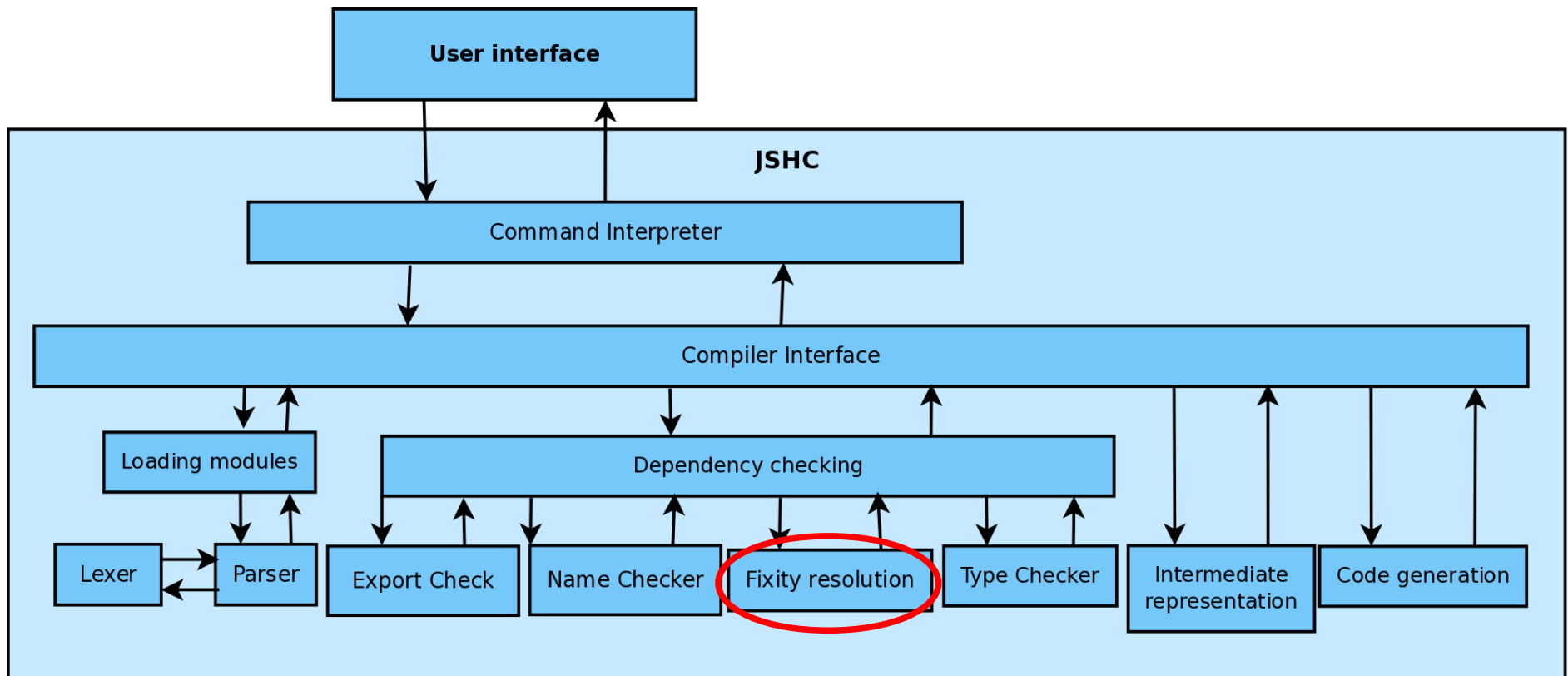
Operatorprecedens

```
infixl 7 *
infixl 6 +
x + y = ...
x * y = ...
```

```
2 * 10 + 3
```

Hur fungerar JSHC?

Operatorprecedens
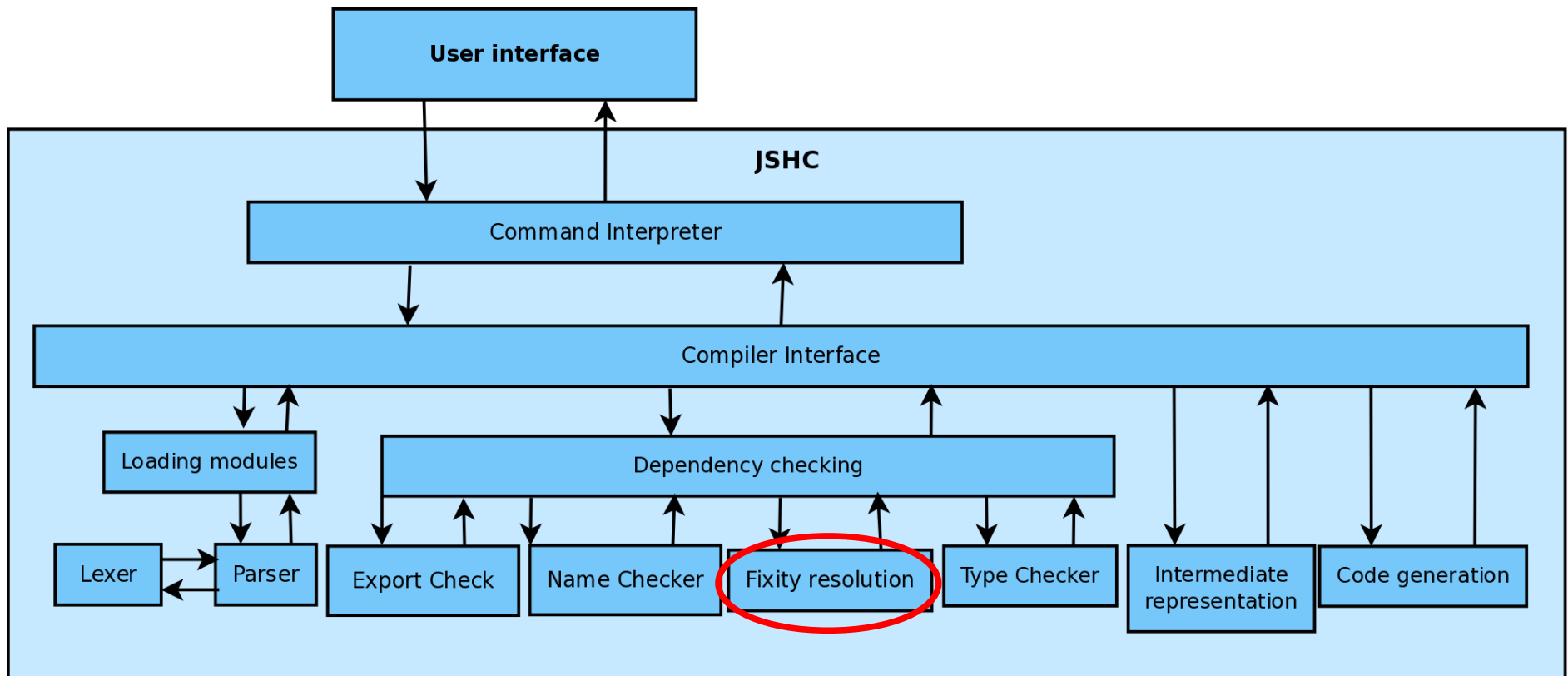
Hur fungerar JSHC?

Typkontroll
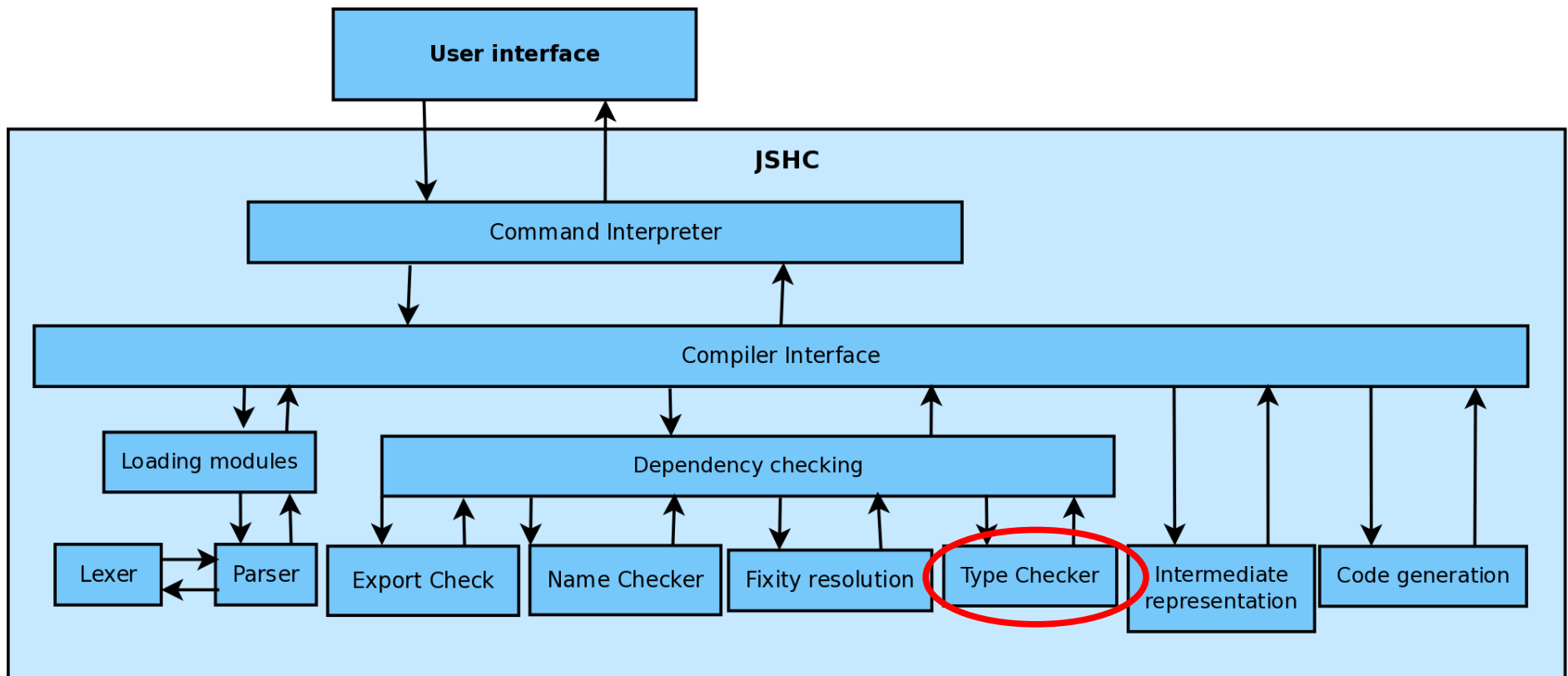
Typkontroll

```
even n = if n==0
  then True else odd (n-1)

odd = if n==0
  then False else even (n-1)
```

Typkontroll

```
map :: (a → b) → [a] → [b]

data Maybe a = Nothing | Just a
Maybe :: * → *
Just :: a → Maybe a
Maybe Int              [Int]
Maybe Maybe            [[]]
```

Hur fungerar JSHC?

Mellanrepresentation
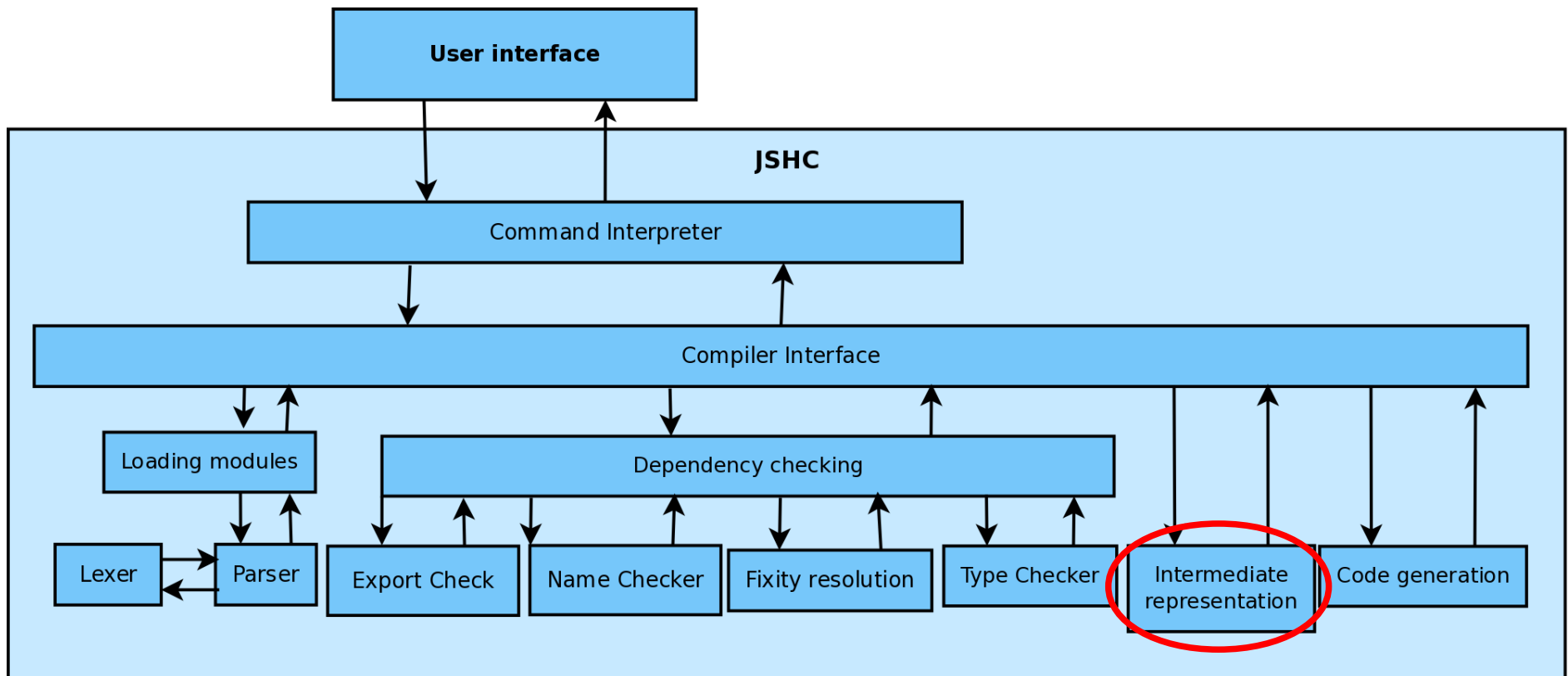
Mellanrepresentation

```
if a < b then a else b

case (<) a b of
  True → a
  False → b
```
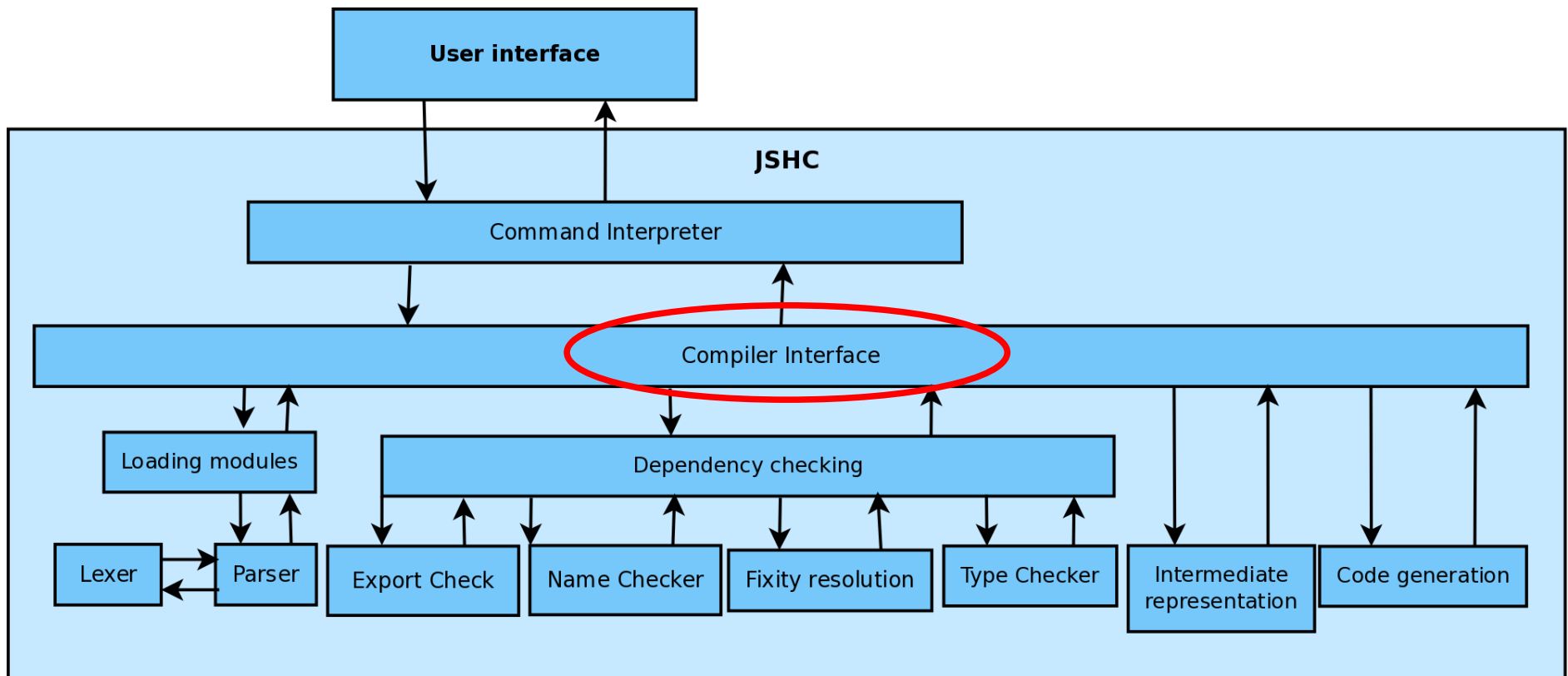
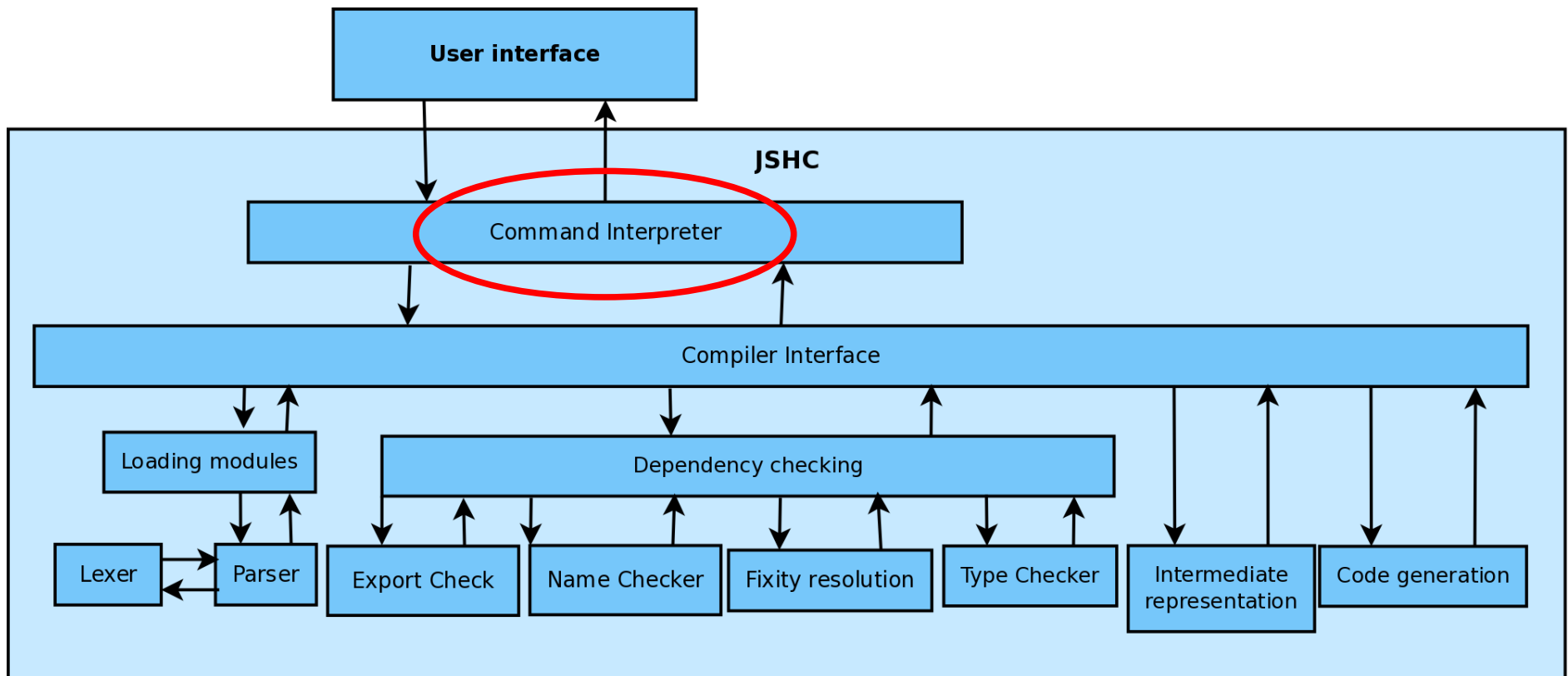# Hur fungerar JSHC?

Generera JavaScript
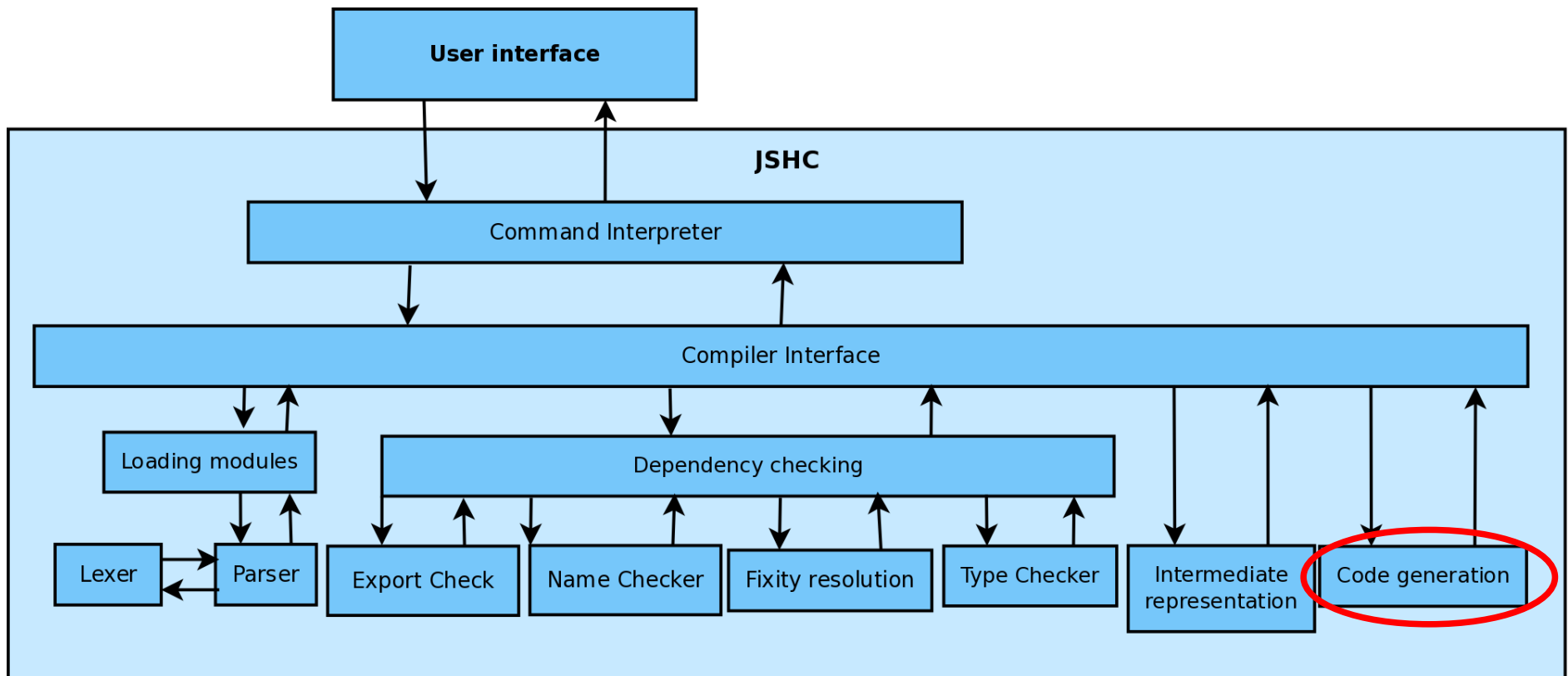
Hur fungerar JSHC?

Exekvera JavaScript

Hur fungerar JSHC?

Kommando tolkas

Problem vi stött på under arbetet

Problem vi stött på under arbetet

Projektets storlek

Problem vi stött på under arbetet

Projektets storlek
- Haskells storlek och komplexitet

Problem vi stött på under arbetet

Projektets storlek
- Haskells storlek och komplexitet
- Dynamisk typning

Problem vi stött på under arbetet

Projektets storlek
- Haskells storlek och komplexitet
- Dynamisk typning

Sinsemellan rekursiva moduler

I framtiden

I framtiden

Typklasser

I framtiden

Fylla ut syntaxen

Typklasser

Foreign function interface

I framtiden

Typklasser

Foreign function interface

Sinsemellan rekursiva moduler

I framtiden

Typklasser

Foreign function interface

Sinsemellan rekursiva moduler

Mindre utökningar av syntaxen

I framtiden

Typklasser

Foreign function interface

Sinsemellan rekursiva moduler

Mindre utökningar av syntaxen

Standardbibliotek