

# Nationaal verkeerskundecongres 2017

## De open verkeersregelaar Op weg naar een open source ITS applicatie voor de iVRI

Harm Jan Mostert  
(Provincie Noord-Holland)

Cyril Cappendijk  
(Cappendijk Verkeersadvies)

Menno van der Woude  
(CodingConnected e.U.)

### Samenvatting

De Provincie Noord-Holland werkt samen met private partijen, kennisinstellingen en weggebruikers aan Smart Mobility toepassingen die de veiligheid, doorstroming en leefbaarheid op haar wegen verbeteren. Hiervoor is de proeftuin Smart Mobility Schiphol ingericht waar innovaties op de provinciale wegen N201, N205, N231 en N232 in de praktijk worden getest.

In samenwerking met CodingConnected e.U. en Cappendijk Verkeersadvies heeft de provincie in dit project getest of de open architectuur voor intelligente verkeerslichten (iVRI) daadwerkelijk kan worden benut door innovatieve nieuwkomers in de verkeersregeltechniek. Enerzijds resulteerde dit in de ontwikkeling van een open source implementatie van het TLC-FI koppelvlak voor ITS applicaties en anderzijds in de ontwikkeling van een nieuwe open source regelstructuur in de programmeertaal C#. Deze zijn inmiddels beide met succes getest op straat en vrij beschikbaar online.

### Trefwoorden

ITS, iVRI, innovatie, open source

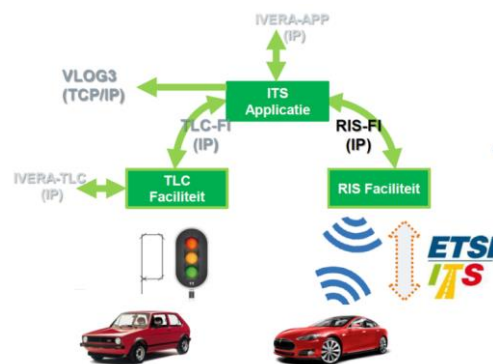
ORGANISATIE 2017



## Aanleiding

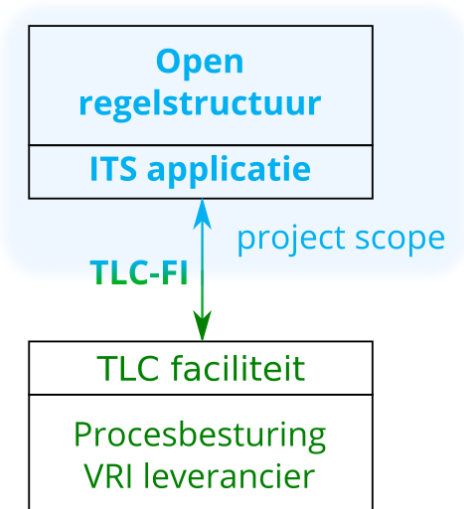
In het kader van het landelijke programma Beter Benutten Vervolg (BBV) is in april 2017 de ontwikkeling van de iVRI architectuur afgerond. De iVRI is een verkeersregelaarsysteem dat informatie extern kan ontsluiten naar de weggebruiker en kan worden aangestuurd door externe systemen van derden. Het aansturen van een iVRI gebeurt door een ITS applicatie via een open interface: de TLC-FI, ofwel Traffic Light Controller Facilities Interface.

De TLC-FI maakt het mogelijk een verkeersregeling te ontwikkelen onafhankelijk van het type automaat en procesbesturing: een externe regelaar kan, lokaal of op afstand (via TCP/IP), verbinding leggen met een iVRI, de actuele status van in- en outputs monitoren, en desgewenst een verzoek indienen voor het overnemen van de controle. Op deze wijze kan de verkeersregeling die op een kruising feitelijk de staat van de lichten bepaalt, op elk gewenst moment worden gewisseld, zonder dat daarbij de leverancier van de automaat betrokken hoeft te worden.



**Figuur 1: iVRI architectuur**

Dit project is erop gericht de openheid van de iVRI



**Figuur 2: Scope van het project**

architectuur aan te tonen en innovatie daarop verder te stimuleren. Focus ligt hierbij op de toepasbaarheid van het TLC-FI koppelvlak door derden. Dit is gebeurd door de TLC-FI aan de applicatiekant te implementeren, en die implementatie te gebruiken voor het koppelen van een nieuwe regelstructuur aan een bestaande TLC faciliteit van een iVRI, in de testomgeving, in de fabriek, en op straat (zie project scope in figuur 2).

Dit project is uitgevoerd in het kader van de proeftuin Smart mobility Schiphol van de provincie Noord-Holland. Hierbij werkt de provincie samen met private partijen aan innovatieve praktijktoepassingen op de provinciale wegen rondom Schiphol. De ontwikkelingen binnen het project zijn uitgevoerd door CodingConnected. Cappendijk Verkeersadvies heeft het project procesmatig begeleid.

## Doelstellingen

Tegen de geschetste achtergrond zijn voor het project de volgende concrete doelstellingen geformuleerd:

- Nagaan of en in hoeverre de TLC-FI als open koppelvlak bruikbaar is voor nieuwe partijen bij de ontwikkeling van ITS applicaties.
- Deze bruikbaarheid aantonen door deze zonder voorkennis van het ontwikkeltraject van de iVRI met een nieuwe, open regelstructuur toe te passen in een iVRI op straat.

## Aanpak

Het project behelst drie deel-ontwikkelingen, die hieronder kort worden toegelicht.

### *Implementatie van de TLC-FI*

Voor de implementatie is aanvankelijk uitsluitend gebruik gemaakt van de documentatie rond de iVRI (IRS en IDD van de TLC-FI). Gedurende de ontwikkeling is een beperkte implementatie van de VRI kant (TLC faciliteit) van de TLC-FI opgebouwd om hiermee het opbouwen van de verbinding en uitwisselen van data te kunnen testen. Op deze wijze is een goed beeld verkregen van de bruikbaarheid van de TLC-FI en bijbehorende documentatie door derden zonder voorkennis van het traject rond de iVRI.

### *Opbouwen van een nieuwe regelaar*

Om de onafhankelijkheid van bestaande technieken te waarborgen, is er voor gekozen geen gebruik te maken van bestaande verkeersregelingen als CCOL of RWS-C. In plaats daarvan is een nieuwe open source regelstructuur opgebouwd. Dit betreft een eenvoudige, maar volledig functionele voertuigafhankelijke module regeling. De huidige versie voorziet tevens in enige flexibiliteit, middels de mogelijkheid voor wachtgroen, meeverlengen en vooruit realiseren.

### *Testen in de praktijk*

De opgebouwde regelaar is voorzien van het TLC-FI koppelvlak, en vervolgens op een eenvoudige kruising getest. Hiertoe is contact gelegd met een aantal fabrikanten van verkeerslichten om de implementatie ook met hun iVRI te kunnen testen. Hiervoor is gebruik gemaakt van testsoftware van Dynniq, wat een uitgebreid testtraject mogelijk maakte. Daarnaast is ter plaatse getest, in de fabriek bij Swarco, en vervolgens op straat in een functionerende iVRI van Swarco.

## Uitvoering

### *C#*

Bij de implementatie is gekozen voor het gebruik van de taal C#, met gebruik van het .NET framework (of mono voor Linux). C# is een moderne taal, die tal van functionaliteiten biedt die de implementatie van de TLC-FI vereenvoudigen, waardoor fouten kunnen worden voorkomen.

De taal vraagt vanwege haar structuur wat meer van hardware dan bijvoorbeeld C. Met moderne hardware of bij regelen "vanuit de cloud" levert dat echter geen problemen op. Het geheel van regelaar met TLC-FI draait goed op een Raspberry Pi versie 1 model B (700 MHz, 512 MB ram, gem. CPU gebruik 5-10%). Gebruik van .NET heeft daarbij als voorbeeld voordeel dat de implementatie van de TLC-FI ook gebruikt kan worden in programma's geschreven in bijvoorbeeld VB.NET of F#.

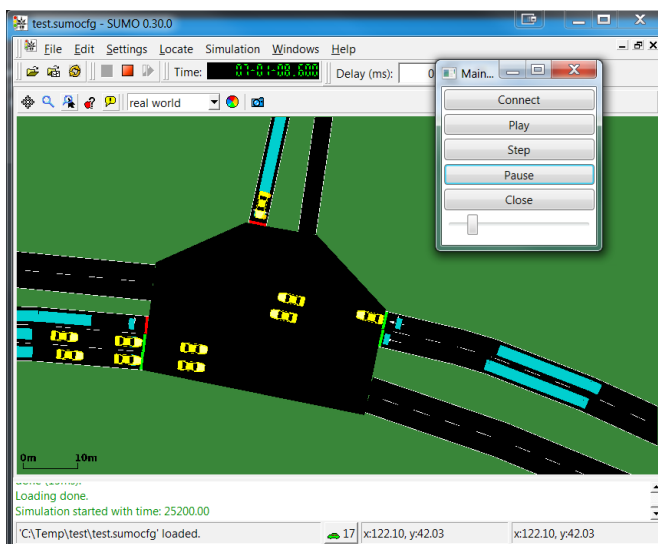
### *Cross platform*

De ontwikkeling is gedaan in het besturingssysteem Windows. Tijdens de ontwikkeling is er daarnaast voor gezorgd dat zowel de implementatie van de TLC-FI als de nieuwe regelstructuur ook bruikbaar zijn onder het besturingssysteem Linux. Dit is tijdens de ontwikkeling voortdurend getest door de software ook te draaien onder Linux ( Debian, Ubuntu en Fedora).

### Cyclisch proces

In de uitvoering zijn de omschreven onderdelen deels parallel uitgevoerd. Dit is gebeurd in een cyclisch proces, waarbij de nieuwe regelstructuur en de implementatie van de TLC-FI steeds completer werden, wat weer uitgebreidere tests mogelijk maakte. Dit leverde grofweg het volgende verloop op in de uitvoering:

- Eerste implementatie van een module regelaar in C#. Deze is ook getest onder Debian Linux op de Raspberry Pi.
- Start met de implementatie van de TLC-FI. Hierbij zijn eerst alle data objecten opgebouwd. Vervolgens is het proces van registratie geïmplementeerd, waartoe ook een beperkte implementatie van de iVRI is opgebouwd.
- Zowel de applicatie kant als de iVRI kant zijn verder opgebouwd, tot het punt dat de applicatie kon verbinden, registreren, en de controle kon overnemen. Dit is eveneens getest onder Linux, door voor beide kanten een Raspberry Pi te gebruiken (zie figuur 3), en het proces van verbinden en uitwisselen van data los van de testomgeving in Windows te laten plaatsvinden.
- Toen de TLC-FI op deze wijze stabiel draaide is gestart met het testen tegen de implementaties van de fabrikanten Swarco en Dynniq. Dit bracht diverse punten naar voren die in de eigen implementatie niet op waren gevallen.
- Ondertussen is ook de regelaar verder ontwikkeld met enige mate van flexibiliteit in het regelproces. Tevens is een testomgeving met bitmap ontwikkeld, en is de regelaar gekoppeld aan het open source microsimulatie pakket SUMO (zie figuur 4).
- Na diverse slagen en test rondes is de regelaar met de TLC-FI succesvol getest in de fabriek en op straat. Hierbij zijn steeds verschillende testopstellingen gebruikt: Windows en Linux, 32 en 64 bits systemen, Intel systemen (pc) en ARM systemen (Pi). Zo is brede inzetbaarheid van de implementatie zoveel mogelijk zeker gesteld.
- De nieuwe regelstructuur wordt in een doorlopend traject getest op correcte en gewenste werking, en met name op praktische bruikbaarheid door verkeersregelkundigen, als alternatief voor bestaande (gesloten) producten.



**Figuur 4: microsimulatie nieuwe regeling met SUMO**



**Figuur 3: testopstelling met 2 Raspberry Pi's (beide met Debian Linux)**

## Resultaten

Het project heeft een succesvol geteste, toepasbare implementatie van de TLC-FI opgeleverd. Daarom kan worden gesteld: de TLC-FI is als open standaard een geslaagd koppelvlak, dat ook door partijen die met iVRI relatief onbekend zijn, goed gebruikt kan worden.

### Bevindingen

Gedurende het project zijn een aantal bevindingen gedaan rond (het werken met) de TLC-FI:

- De documentatie is overzichtelijk, op veruit de meeste punten eenduidig, en is daarom een goede basis voor de implementatie van de TLC-FI. Komen tot een concrete implementatie vergt wel grondige kennis van die documentatie, en voor sommige aspecten, zoals het proces van registratie, moet de benodigde informatie op diverse plaatsen in de documentatie worden verzameld.
- Er bestaan kleine verschillen tussen de implementaties van de TLC-FI aan de zijde van de iVRI. Die zorgden tijdens het testen van de implementatie soms voor onverwachte foutmeldingen. Het betreft echter relatief kleine punten, en de TLC-FI biedt dan ook een goede basis voor interoperabiliteit tussen de iVRI en externe regelapplicaties
- Gezien de centrale rol van uitwisseling van data bij de TLC-FI, zijn test tools, waarmee dat proces van communicatie kan worden nagebootst, onontbeerlijk. Testen met een eigen tool levert daarbij onvoldoende inzicht op om een nieuwe implementatie volledig te testen.

### Producten

Aangezien de toepasbaarheid van de TLC-FI binnen het project is getest door het ontwikkelen van een praktische implementatie, heeft dit ook een aantal concrete producten opgeleverd. In aansluiting op het open karakter van de iVRI, is er daarbij vanaf de start voor gekozen de resulterende broncode na afronding van het project open source te ontsluiten.

Het project heeft de volgende concrete producten opgeleverd:

- Open source implementatie van de TLC-FI (applicatie zijde) in C#. Deze is online beschikbaar: <https://www.codingconnected.eu/software/tlcfi-net/>.
- Nieuwe, open source regelstructuur, eveneens in C#, en eveneens online beschikbaar: <https://www.codingconnected.eu/software/tlcprof/>.
- De mogelijkheid de nieuwe regelaar in een micro simulatie te simuleren met SUMO (een open source pakket voor microsimulaties). Hiertoe is een kleine (open source) functie bibliotheek ontwikkeld: <https://github.com/CodingConnected/CodingConnected.Traci>
- Gedurende het traject is steeds gezorgd dat de TLC-FI en nieuwe regelaar tevens onder Linux kunnen draaien. Dit maakt ook het gebruik op een klein formaat pc zoals bijvoorbeeld de Raspberry Pi mogelijk.
- De in dit project ontdekte interpretatie verschillen worden met de beheerder van de koppelvlakspecificaties gedeeld, zodat verduidelijkingen in de documentatie kunnen worden aangebracht.

## Stand van zaken en vervolg

Het project bevindt zich inmiddels in de afrondingsfase. De implementatie van de TCL-FI is getest in de fabriek en op straat, en heeft in beide situaties een iVRI aangestuurd. De broncode is inmiddels in bèta vorm online beschikbaar voor gebruik en testen door anderen. Dit geldt eveneens voor de nieuwe regelstructuur. De implementatie van de TLC-FI is vrijwel volledig, echter ten opzichte van het ontwerp van de iVRI en de rol daarin van de ITS applicatie, ontbreken nog een aantal aspecten. Dit betreft bijvoorbeeld de mogelijkheid de regelaar te monitoren en aan te sturen vanuit beheer- en verkeerscentrales.

De provincie Noord-Holland wil een dergelijke open ITS applicatie samen met andere overheden en kennisinstellingen verder doorontwikkelen, zodat er naast de gesloten implementaties van private partijen ook een volledig open alternatief komt voor het regelen van kruispunten. Dit stimuleert innovatie voor het slimmer regelen van kruispunten en kan ook gebruikt worden in praktijkgericht onderzoek en onderwijs. Met de komst van iVRI en de ontwikkeling en testen die in dit project zijn uitgevoerd is aangetoond dat dit mogelijk is.

De komende maanden zullen CodingConnected e.U. en Cappendijk Verkeersadvies de basis van de open source regelstructuur verder uitwerken en optimaliseren. Daarbij worden tevens de documentatie en specificaties voor de reeds geïmplementeerde functionaliteit op orde gebracht. In een volgend stadium worden ook bijzonderheden zoals nalopen, gelijk- en voorstarten en file- en ov-ingrepen uitgewerkt. Hierbij wordt zoveel mogelijk aansluiting gezocht bij de in Nederland op grote schaal toegepaste functionaliteiten in CCOL, RWS-C of de (vrij beschikbare) CCOL Generator van de provincie Noord-Holland. In dit traject wordt expliciet samenwerking gezocht met de regeltechnische vakwereld, om te komen tot een breed gedragen structuur.

## Conclusie

Op basis van de resultaten van de testen in de fabriek en op straat is aangetoond dat de iVRI architectuur kan worden benut door nieuwe partijen die ITS applicaties ontwikkelen. Zowel softwarematig als hardwarematig is er onafhankelijkheid aangetoond door de toepassing van een nieuwe regelstructuur. Hiermee zijn de innovatiemogelijkheden van verkeersregelingen verder vergroot.

‘Leren door te doen’ en ‘zien is overtuigen’ stonden centraal binnen het project. Innovatieve projecten kennen vooraf aan de uitvoering veel onzekerheden, omtrent scope, planning, en praktische uitvoerbaarheid. Door gewoon te starten, en bij elke stap te evalueren en bepalen hoe verder, heeft het project relatief snel tot concrete resultaten geleid.